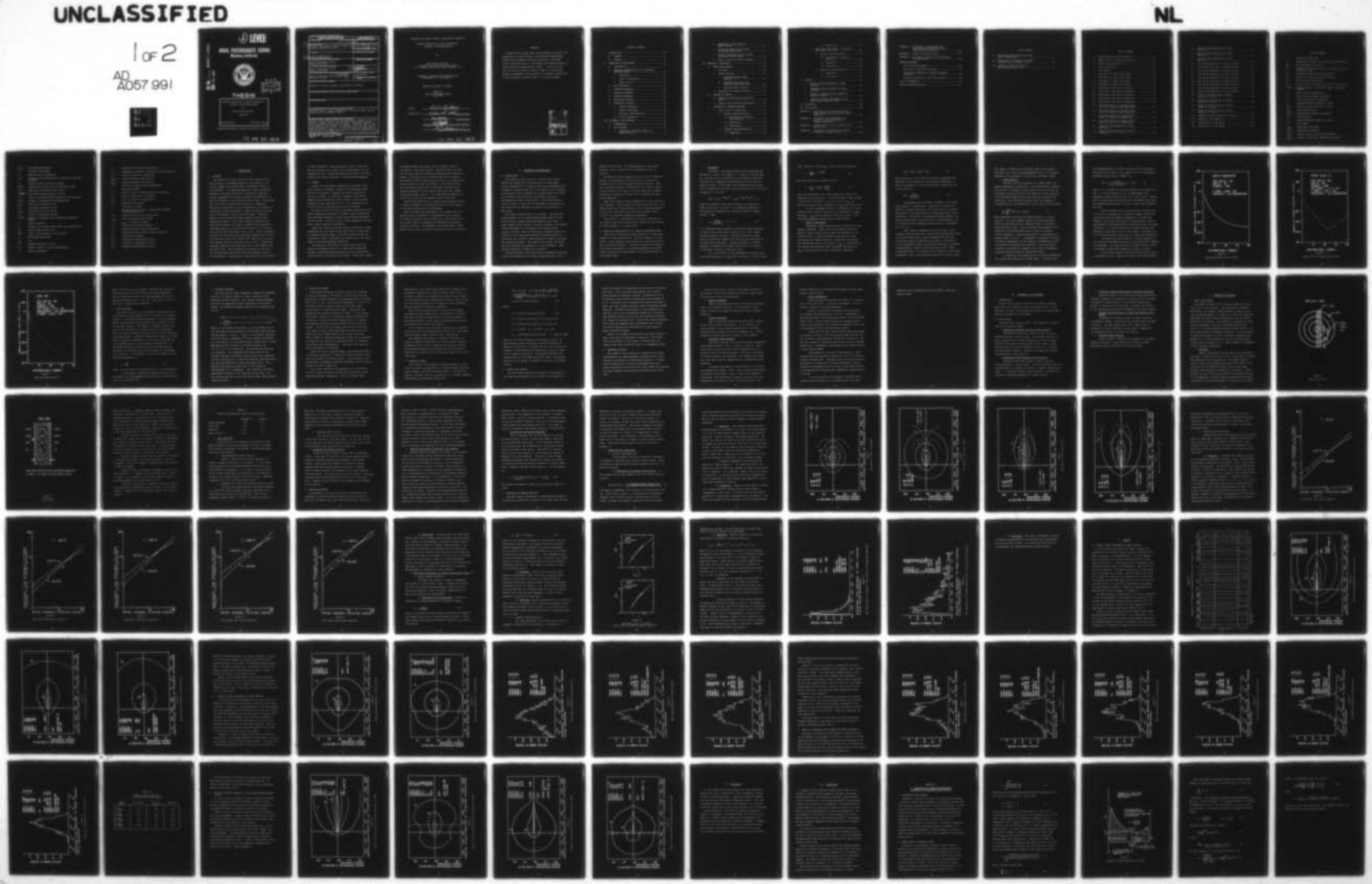
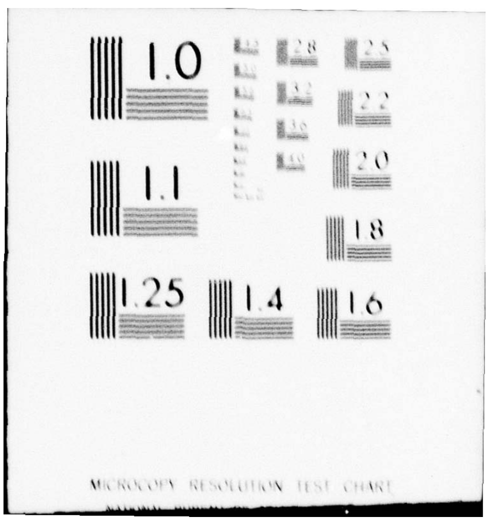
AD-A057 991 NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF F/G 20/6
COMPUTER SIMULATION OF LIGHT PROPAGATION THROUGH A SCATTERING M--ETC(U)
JUN 78 M A MILLBACH

UNCLASSIFIED NL

1 OF 2

AD
A057 991




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
MICROCOPY RESOLUTION TEST CHART

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

Computer Simulation of Light Propagation
through a Scattering Medium

by

Miles Allen Millbach

June 1978

Thesis Advisor: William M. Tolles

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER 9
4. TITLE (and Subtitle) 6 Computer Simulation of Light Propagation through a Scattering Medium.		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis, June 78
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) 10 Miles Allen Millbach		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey CA 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey CA 93940		12. REPORT DATE 11 June 1978
		13. NUMBER OF PAGES 160
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School MOnterey CA 93940 12 160p.		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Multiple Scatter, Light Propagation, non-line-of-sight, Optical Communication, Atmospheric Optics.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A Monte Carlo computer model was developed to simulate the propagation of light through a scattering/absorbing medium using various parameters and phase functions. The model permits characterization of the spatial and temporal spread of light traversing plane-parallel clouds. It was found that both the time and spatial spread of light in a scattering medium are independent of the details of the phase function for a cloud thickness of greater than 15 extinction lengths.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

251 450 LB

Approved for public release; distribution unlimited

Computer Simulation of Light Propagation through a Scattering Medium

by

Miles Allen Millbach
Lieutenant, United States Coast Guard
B.S., United States Coast Guard Academy, 1973

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN PHYSICS

from the

NAVAL POSTGRADUATE SCHOOL
June 1978

Author: Miles A. Millbach

Approved by: William M. Tolles, Thesis Advisor

A. W. Cooper, Second Reader

K. E. Woehler, Chairman, Department of Physics and Chemistry

G. J. Haltiner, Dean of Science and Engineering

ABSTRACT

A Monte Carlo computer model was developed to simulate the propagation of light through a scattering/absorbing medium using various parameters and phase functions. The model permits characterization of the spatial and temporal spread of light traversing plane-parallel clouds. It was found that both the time and spatial spread of light in a scattering medium are independent of the details of the phase function for a cloud thickness of greater than 15 extinction lengths.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

a =	absorption coefficient
a =	(Appendix B) constant of particle size distribution
a_n =	Mie scattering coefficient
A_i =	(Appendix A) intercept of line connecting (θ_i, P_i) and (θ_{i+1}, P_{i+1})
A_j =	(Appendix B) Mie scattering amplitude
b =	constant of particle size distribution
b_n =	Mie scattering coefficient
B_i =	slope of line connecting (θ_i, P_i) and (θ_{i+1}, P_{i+1})
$\langle\cos\theta\rangle$ =	average cosine of the scatter angle θ for phase function
C_i =	factor used in generating θ_R
D =	mean free path between collisions
F =	total flux through an aperture
G =	Henyey-Greenstein parameter = $\langle\cos\theta\rangle$
h_n^2 =	spherical Bessel function of order n
H_B =	beam spread function
$i_j(\theta)$ =	dimensionless intensity parameters
I =	beam intensity
j_n =	spherical Bessel function of order n
k =	wave number
K_{abs} =	absorption coefficient
K_{ext} =	extinction coefficient
K_{Mie} =	portion of K_{sca} due to Mie scattering
K_{Ray} =	portion of K_{sca} due to Rayleigh scattering

K_{sca} = scattering coefficient

L = multipath time spread

m = complex index of refraction

n(x) = particle size distribution as a function of Mie size parameter

N = number of phase function data pairs

NORM = normalization factor used for individual panel

$P(\theta)$ = normalized scattering phase function

$P(\theta,\phi,\theta',\phi')$ = Normalized scattering phase function

P_i = value of scattering phase function at ith data pair

$P_j(\theta)$ = elements of normalized scattering matrix

Q_{ext} = extinction efficiency factor

Q_s = scattering efficiency factor

Q_{sca} = scattering efficiency factor

r = radius of scattering particle

$\bar{r},\bar{r}',\bar{r}''$ = spatial coordinates, time adjusted for numerical method

R = range of distance

R = (Appendix A) uniform random number in interval [0,1]

R_1 = random number related to R

RPT = ratio of particulate to total scatter

s = scattering coefficient

S_j = kA_j

T = physical thickness of cloud

W_i = weight of panel i, used in generating θ_R

x = Mie size parameter

α = (Appendix E) extinction coefficient

α = (Appendix B) constant of particle size distribution

α_e = effective extinction coefficient

β_{ext} = volume extinction cross section

β_{sca} = volume scattering cross section

γ = constant of particle size distribution

γ_o = RMS scatter angle

ε = absolute value of index refraction of particle

θ = angle off direction of incidence

θ_i = value of theta in ith data pair

θ_R = weighted scatter angle

λ = wavelength of light

μ = absolute value of index of refraction of medium surrounding particle

π_n = angular functions used in Mie Series

σ_{sca} = total scattering cross section

τ = $K_{ext}\ z$ = optical thickness

τ_d = effective scattering thickness

τ_n = angular functions used in Mie Series

ϕ = longitudinal angle around incident direction

ω_o = single scatter albedo

Ω_x = coordinate transformation factor

Ω_y = coordinate transformation factor

Ω_z = coordinate transformation factor

I. INTRODUCTION

A. HISTORY

The existence of clouds and fog in many regions of the earth presents a formidable problem to the designer of an optical communication system whose transmission channel is the atmosphere. A scattering medium can inhibit system performance by inducing beam spread, dispersion in angle-of-arrival, degradation of spatial coherence and dispersion in time and frequency of the modulated optical beam. The development of an optical communication system for atmospheric applications requires an accurate knowledge of the effects of scattering on light propagation. Numerous studies have been made in an attempt to provide this knowledge. Different aspects of the problem have been assaulted using various mathematical models. Monte Carlo computer simulation has been used by Bucher [1], Plass and Kattawar [2] through [5], Junge [6] and Danielson, Moore and van de Hulst [7], to mention only the few used extensively in this work. Equally prominent attempts have been made using analytical methods such as those by Arnush [8], Gordon [9], Stotts [10], Hansen [11], Ishimaru [12], Kennedy [13] and Lutomirski and Yura [14]. Complicated numerical techniques have been used only by Dell-Imagine [15] and Zachor [16]. Each attempt has contributed to the stockpile of knowledge required but additional problems remain to be investigated. Many books have been published on the subject

of light scattering. Those found most useful to this work were van de Hulst [18], Kerker [19], Chandrasekhar [20] and Deirmendjian [21]. Interested parties and reasons for their interest are briefly discussed in the following paragraphs.

B. PURPOSE

Current navy operational communications systems suffer from a number of problems. There is no operational system which is not subject to jamming, intercept, spoofing and direction finding. In addition, it appears that optical communications systems have great promise in solving these problems for many applications [17]. Any information that could be used in evaluating such systems is desired.

The Navy's effort to create a worldwide satellite-to-submarine communication network is another rapidly progressing area requiring information of the nature being investigated in this thesis. In this situation, light propagation through water further complicates the matter.

Among other parties that may require information on light propagation are the United States Coast Guard in their aids to navigation system, NASA in satellite communications and researchers trying to determine the composition of the atmosphere by analyzing scattered radiation.

The purpose of this work is to characterize both the spatial and temporal aspects of a light beam propagating through a scattering/absorbing medium using an analytically verified Monte Carlo model of the system. Models were created

and tested against each other until an adequate level of confidence in each was obtained. The models were then used to study more specific questions concerning time and spatial spread. More explicitly, the models were used to observe what effect different phase functions had on the temporal and spatial aspects of light scattering. It was believed that the pronounced forward peak of some phase functions would generate different time and spatial character than a moderately peaked phase function. An investigation attempting to verify this belief was an important part of this work.

Spatial characterization of light passing through a cloud of finite thickness, and transition from the forward scatter to diffusion region was also investigated.

Once confidence was established in the models used, the general areas of agreement or disagreement were studied in an attempt to specify when one model may be used more efficiently than another for generating information concerning a specific situation. The drawbacks and strong points of each model type have been determined and mentioned where appropriate.

II. THEORETICAL PRELIMINARIES

A. INTRODUCTION

The purpose of this section is to review the various theoretical formulations currently used to characterize optical propagation in a single and multiple scattering medium. A complete coverage of any one of the topics in the following sub-sections is most certainly not contained here but sufficient references are provided for the reader inclined to investigate any topic in more detail. Only those details related closely to this work are described in any detail and even then only the important conclusions are mentioned in many cases.

The order of presentation is as follows. The principal characteristics of the scattering/absorbing medium are mentioned briefly, thereby defining frequently used terms. A brief summary of Mie theory is included to present the valuable insight necessary in investigating the problems posed. Radiative transfer theory is mentioned in passing because the solution of the complicated radiative transfer equation is essentially the subject of many analytical attempts at solving the scattering problem. This section then probes into mathematically modeling the geometry of a real atmosphere. Three general categories of models are discussed.

Analytical models, which made various assumptions allowing closed form expressions for certain characteristics of the

medium, are discussed. The approximations in such models normally detract from the total worthiness of such an approach.

The discussion then progresses into the use of numerical methods in evaluating analytical solutions as well as in evaluating the radiative transfer equation directly. This often overlooked powerful technique has been used by theoreticians because of its ease in solving complicated integro-differential equations for which exact solutions have been proven impossible.

Monte Carlo methods, used predominantly in this work, are then discussed. This technique is used extensively due to its ease in adaptation to odd geometries of the scattering medium. In this work this benefit was used only to model finite clouds in an otherwise homogeneous atmosphere.

This section is completed with a general discussion of the results predicted by theoretical insight into the problem of light transfer in a random medium.

B. PRINCIPAL CHARACTERISTICS OF SCATTERING/ABSORBING MEDIA

This section divides the characteristics of a scattering medium into three sub-sections. The first defines the parameters of the atmosphere which during the course of this work were considered constants. The second considers the types of scattering encountered in a scattering medium and the third briefly describes the significance of the phase function in scattering theory.

1. Parameters

In general, a medium may exhibit both scattering and absorption. Some of the energy of a collimated incident beam on a particle will be scattered over all possible directions; the rest will be absorbed by the particle and lost from the radiation field. In the single scattering problem the intensity of radiation at any point along the direction of plane wave propagation is given by:

$$I(R) = I(0)e^{-(K_{ext})R} = I(0)e^{-(K_{abs}+K_{sca})R}, \quad (1)$$

where the absorption loss and scattering loss together is called extinction. The ratio of scattering coefficient to the extinction coefficient is called the albedo of single scatter. Thus,

$$\omega_o = \frac{K_{sca}}{K_{abs}+K_{sca}}, \quad 0 < \omega_o < 1 \quad (2)$$

It is sometimes more convenient to discuss optical dimensions in terms of mean free paths of photons, i.e., the average distance between collisions. The mean free path is often called an extinction length and is given by the reciprocal of the extinction coefficient. Optical thickness of a given medium is a dimensionless number representing its thickness in extinction lengths or mean free paths.

The transmittance of the atmosphere is the fractional intensity remaining in a beam after traversing a path R units

long. Explicitly, with K_{ext} in km^{-1} and R in kilometers,

$$\tau = \frac{I(R)}{I(0)} = e^{-K_{ext}R} . \tag{3}$$

The term visibility is defined as [22]

$$V = \frac{1}{K_{ext}} \ln \frac{1}{.02} = \frac{3.912}{K_{ext}} , \tag{4}$$

thus the transmittance for a path length just equal to the visibility is two per cent. In many atmospheric light propagation problems the albedo is close to unity and K_{ext} in the above equations can be expressed by K_{sca} with little error.

In a real atmosphere there is a contribution to scattering by small particles as well as a contribution by large particles. K_{sca} then is defined in even more detail. The following paragraphs present some of these details.

2. Types of Scattering

Atmospheric light scattering can be classified in two general categories. Small particle scattering, where the particle radius is much scaller than the wavelength of the incident light, is called Rayleigh scattering, and other scattering from larger particles is termed Mie scattering. The terminology of this work refers to K_{Ray} as the portion of the scattering coefficient due to Rayleigh scattering and K_{Mie} as the portion due to Mie scattering. Therefore,

$$K_{ext} = K_{abs} + K_{Mie} + K_{Ray}, \tag{5}$$

Mie scattering is often called particulate scattering while Rayleigh is often referred to as molecular scattering. A ratio of particulate to total scattering is defined for use in this thesis as:

$$RPT = \frac{K_{Mie}}{K_{Mie}+K_{Ray}} . \tag{6}$$

Scattering may be isotropic (scatters in all directions equally) or anisotropic (scatters as a function of angle off incident direction). The latter only is used here. The problem of defining scatter for single particles is well understood. The theory of scattering has been extended to a medium of many equal sized particles, a monodispersion, and to a medium of many different sized particles, a polydispersion [18, 21].

When a photon undergoes only one collision in traversing a scattering medium, single scattering has taken place. On the other hand, when a photon undergoes more than one collision, multiple scattering has occurred. This work involves independent single and multiple scattering in spherical polydispersions, where no scattering event affects other scattering events. For further details related to scattering parameters, types of scattering and other more specific topics

the reader is referred to an excellent text on the subject, McCartney [22]. The following paragraphs focus on the angular dependence of scattering for scattering of different types.

3. Phase Function

The phase function expresses in a formal manner the angular dependence of scattering. The phase function, denoted here by $P(\theta)$, is defined by van de Hulst [18] as the ratio of the energy scattered per unit solid angle in a given direction to the average energy scattered per unit solid angle in all directions. This definition requires that the integral of the phase function be normalized to unity, which is to say that,

$$\frac{1}{4\pi}\int_0^{2\pi}\int_0^{\pi} P(\theta)\,\sin\theta\,d\theta d\phi = 1 \qquad (7)$$

When expressed in this way the phase function is a scalar function. In a more complex analysis of scattering theory, there are phase functions of different types for different polarizations of light. These functions form elements of the normalized scattering matrix which is used to transform Stoke's parameters. Detailed discussion of the parameters is given by Refs. 18 and 21. In Appendix B a program to evaluate the Stoke's parameters for the spherical polydispersions used in this work is presented and explained. The scalar phase function is obtained by averaging the two Stoke's parameters obtained using this computer adapted Mie theory.

In many cases the phase function is easily approximated as a closed form function of scatter angle. In these cases,

the Henyey-Greenstein function is often used as the approximating function because of its simple form. For the Henyey-Greenstein phase function (Figure 1),

$$P(\theta) = \frac{(1-G^2)}{4\pi(1+G^2-2G\cos\theta)^{1.5}}, \quad -1<G<1 \qquad (8)$$

where G is a parameter which equals $\langle\cos\theta\rangle$ for the phase function. A G value between .80 and .87 approximates real atmospheric phase functions quite well in most cases. However, the following phase functions were not represented well using the Henyey-Greenstein phase function no matter what parameter was used.

Phase functions used in this work were of Water Cloud C.2 [21] at a wavelength of .53 microns (Figure 2) which was calculated using the program of Appendix B, and NOSC Fog (Figure 3) which was calculated in a similar manner using an experimentally determined particle distribution near San Diego, California. For further discussion of phase functions, see "Methods of Simulation" later in this thesis. References 18 and 21 study the phase function and its generation extensively.

In general the phase function is quite symmetric in the forward and backward direction for particles small compared to wavelength. For large particles the phase function has an increasingly complicated dependence on angle of scatter.

As one might imagine, the scalar phase function plays a significant role in single and multiple scattering problems.

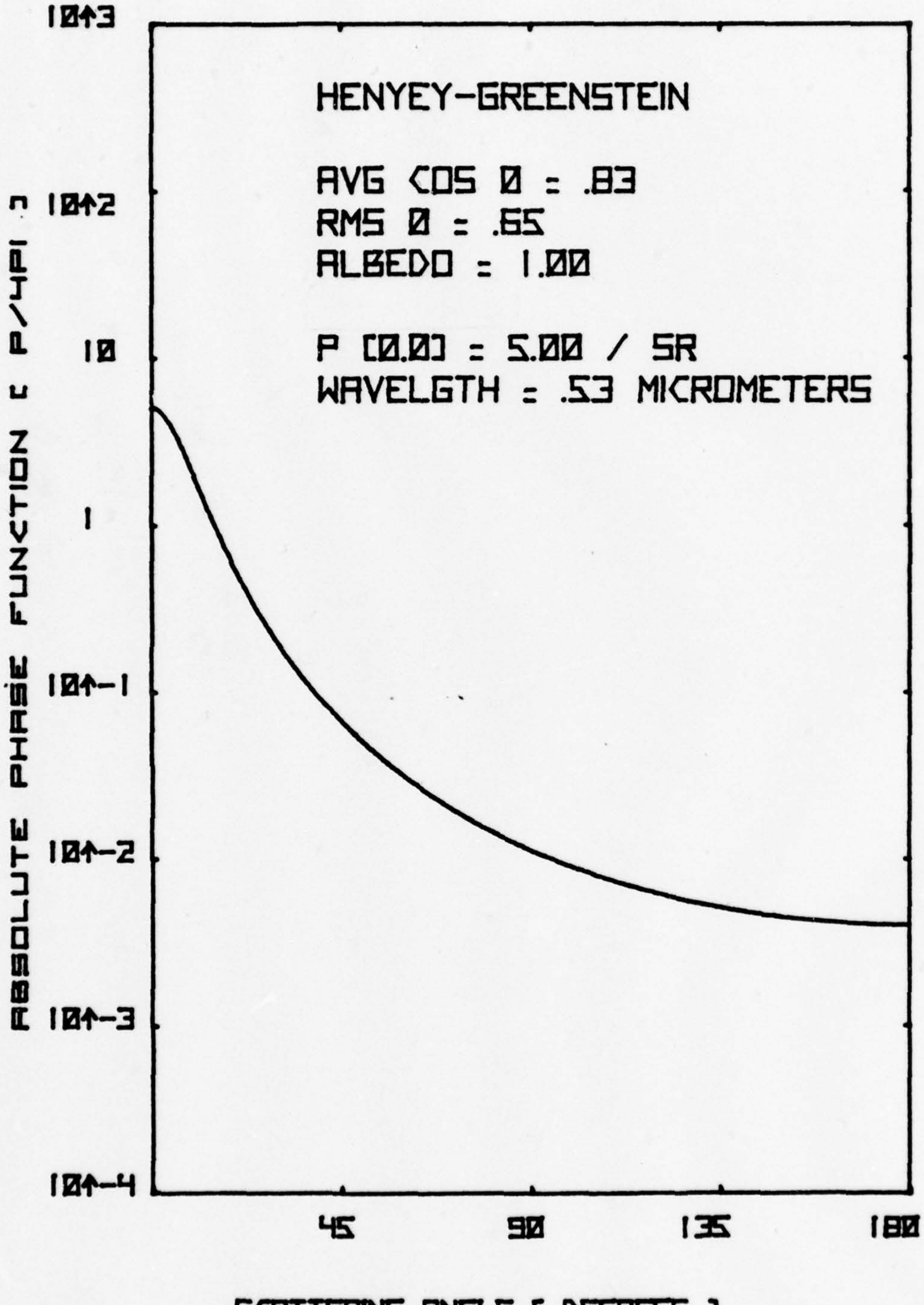


Figure 1

Henyey-Greenstein Phase Function

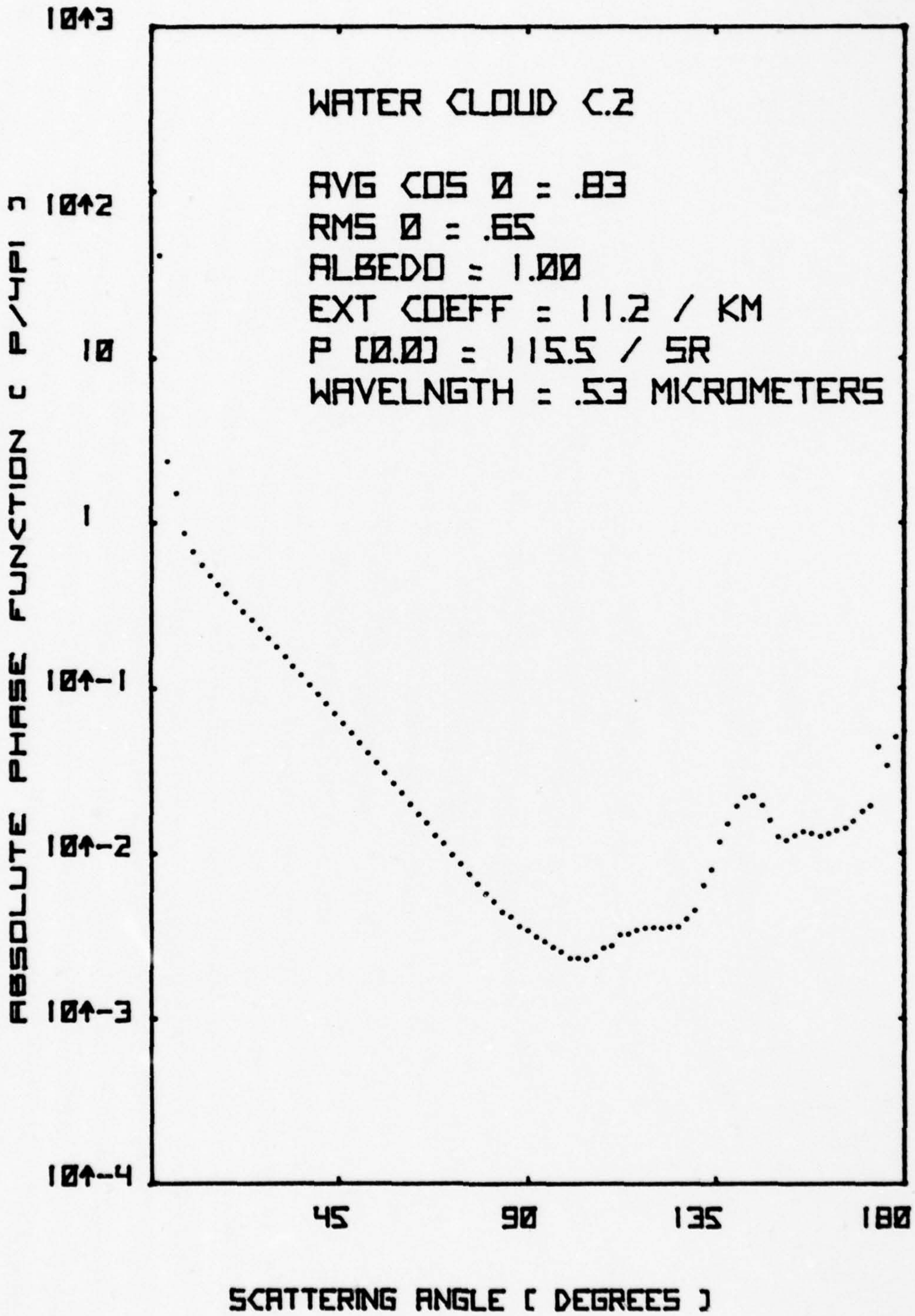


Figure 2

Water Cloud C.2 Phase Function [21]

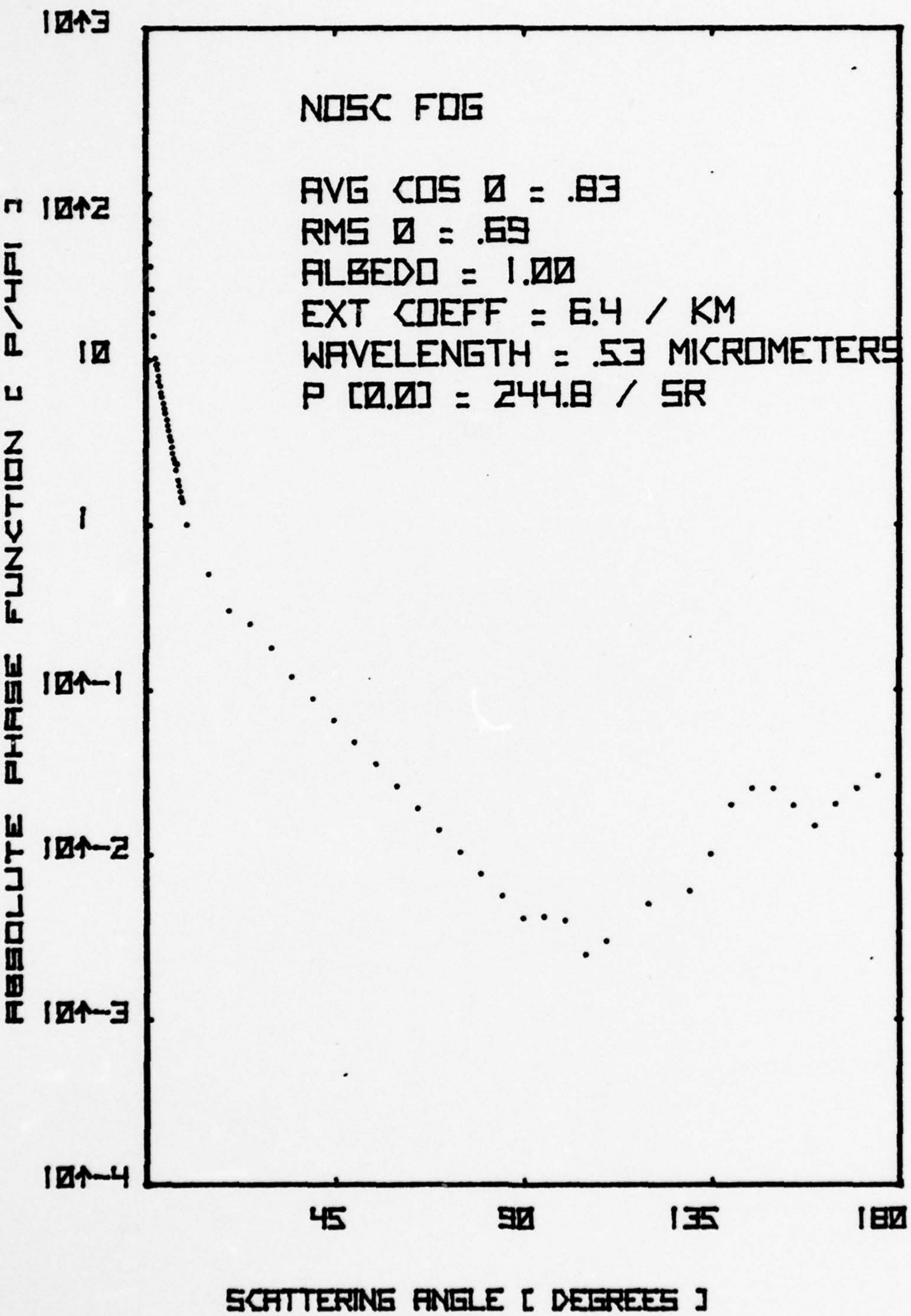


Figure 3

NOSC Fog Phase Function

Much attention has been focussed on defining the conditions under which the entire Mie series can be approximated by analytic expressions such as the Henyey-Greenstein function mentioned previously [1, 7, 11, 18]. This thesis is part of the quest to define these conditions.

C. MIE SCATTERING

Mie theory considers the scattering and absorption which occurs when an electromagnetic wave is incident on a spherical particle. It generates desired parameters of the atmosphere, namely the extinction and scattering cross sections, as well as the phase function. Mie theory is extremely versatile in that it can be applied at any particle size to wavelength ratio and can be extended to many particles and more importantly polydispersions. Appendix B explains the adaptation of Mie theory to machine computation for collecting scattering data for polydispersions of a given particle distribution. As mentioned before, Mie theory was used to generate optical parameters of the Water Cloud C.2 [21] model at a wavelength of .53 microns. It is prudent at this time to define the Mie scattering parameter as

$$x = \frac{2\pi r}{\lambda} \tag{9}$$

where r is the radius of the particle and λ is the wavelength.

For the reader interested in complete derivations of Mie theory, see Chandrasekhar [20], Sekera [23], Born and Wolf [24] or van de Hulst [18].

D. RADIATION TRANSFER

Any discussion of light propagation through the atmosphere would not be complete without at least a mention of the radiative transfer equation. This equation models mathematically what is happening as light transverses a scattering medium. The radiative transfer equation in its simplest form is [10],

$$(\mu \frac{d}{d\tau} + 1)\, I\, (\tau, \bar{r}, \mu, \emptyset, t) = \omega_o I_o\, (\tau, \bar{r}, \mu, \emptyset, t)$$

$$-\omega_o \int_0^{2\pi}\int_{-1}^{1} P(\theta, \emptyset; \theta', \emptyset')\, I\, (\tau, \bar{r}, \mu', \emptyset', t)\, d\mu' d\emptyset' \qquad (10)$$

where I is the scattered radiance, I_o is the radiance due to the distributed source produced by the incident beam transversing the region, $\mu = \cos\theta$, $\tau = K_{ext}\, z$ is the optical thickness, $\bar{r}$ are the spatial coordinates, θ, $\emptyset'$ are angular coordinates, ω_o is the albedo, P is the scalar phase function and t is time. Approximations can be made to solve this equation for $I(\tau, \bar{r}, \mu, \emptyset, t)$ in closed form. The equation has basically two components. One represents the field caused by the incident spatially distributed field and the other represents the field scattered out of the direct beam but redirected back into the same direction by other scattering events. For a complete derivation and explanation of this equation, Chandrasekhar [20] is suggested. Under specific conditions, approximate solutions to this equation can be found. The next section mentions a few of the efforts made toward finding useful solutions.

E. ANALYTICAL METHODS

As mentioned previously, several authors have attempted to provide knowledge of the effects of particulate multiple scattering on light propagation by applying analytical methods [8-13, 20, 25, 26]. One analytical development by Arnush [8] utilized radiative transfer theory and the small angle approximation to characterize the light. Arnush made two assumptions that greatly weakened his model: (1) He assumed the incident signal did not experience pulse broadening while traversing an optically thick medium, and (2) he assumed the incident beam never directly created internal emission sources. Stotts [10] does consider the previous details but still makes the small angle approximation, claiming that the phase function is highly peaked in the forward direction. Other attempts at gathering knowledge include that by Ishimura and Hong [12] who reported an analytical study of coherence using first order approximations. The results are valid for weak fluctuations in the medium.

Zachor [16] uses a double-integral transform method which is evaluated recursively to obtain the aurole radiances contributed by successive scattering orders. He has assumed in his calculations a homogeneous unbounded atmosphere and his results are good for short ranges only.

In any case, because of the complexity of the integrals involved, exact analytical methods yield results only after repeated numerical integration. Even in the small angle

approximation, the closed form solution [27] requires seven successive integrations to obtain a single value of the irradiance caused by a unidirectional point source of light. All this numerical work, besides being tedious, tends to mask the functional dependence of the results on the underlying physical geometry and optical parameters.

There are many complications in solving the multiple scattering problem analytically. Nevertheless, analytical work does contribute significantly to overall knowledge of the problem because in many cases the approximations made do not substantially deviate from reality.

In this thesis, two of the more straightforward analytical methods are used; Gordon [9] and Stotts [26]. The former concerns spatial spread and the latter, time spread.

Dell-Imagine [15] derives a solution to the radiative transfer equation analytically in the usual fashion using no assumptions until he has to actually get numerical results. His numerical approach to evaluating the transfer equation seems quite attractive and is briefly mentioned in the following section.

F. NUMERICAL METHODS

The solution of the radiative transfer equation is sufficient to specify the properties of a received signal which has passed through a multiple scattering region. The solution, however, requires numerical computation on a digital computer. The solution has the general form [15],

$$I(x,y,z,t,\theta,\emptyset) = I(\bar{r},\theta,\emptyset,0)\exp\left[-Q_s c\int_0^t n(\bar{r}'')d\lambda\right]$$

$$+ \frac{Q_s c}{4\pi}\int_0^t\int_0^{\pi}\int_0^{2\pi} n(\bar{r}'')\exp\left[-Q_s c\int_0^t n(\bar{r}')d\gamma\right] I(\bar{r}'',\theta',\emptyset',\lambda)$$

$$P(\theta,\emptyset;\ \theta',\emptyset')\sin\theta' d\theta' d\emptyset' d\lambda \qquad (11)$$

where,

$$\bar{r} = \{(x-\Omega_x ct),(y-\Omega_y ct),(z-\Omega_z ct)\} \qquad (12)$$

$$\bar{r}' = \{(x-\Omega_x c(t-\gamma)),(y-\Omega_y c(t-\gamma)),(z-\Omega_z c(t-\gamma))\}$$

$$\bar{r}'' = \{(x-\Omega_x c(t-\lambda)),(y-\Omega_y c(t-\lambda)),(z-\Omega_z c(t-\lambda))\}$$

$\Omega_x = \sin\theta\cos\emptyset \quad \Omega_y = \sin\theta\sin\emptyset \quad \Omega_z = \cos\theta$

Q_s = scattering efficiency factor   c = speed of light

which has not been restricted to any shape of cloud. By establishing boundary conditions, initial conditions and density of particles n, a solution for $I(x,y,z,t,\theta,\emptyset)$ can be obtained by approximating the equation by a group of simultaneous algebraic equations. Dell-Imagine describes this approximation and the numerical methods used in solving the equation and the reader is referred to his work for further details.

G. MONTE CARLO METHODS

The Monte Carlo method can be applied to any problem if one knows the probability for each step in the sequence of

events and desires the probability of the total of all possible events. Thus, the Monte Carlo method may be used to study problems in radiative transfer. As mentioned in the introduction, many authors have described their attempts to do so. The Monte Carlo calculations are relatively easier to model than other methods especially when the geometry becomes difficult. The Monte Carlo method is very consumptive of computer time and only approximate information is obtained. Some generally useful results have been presented by Bucher [1], Junge [6] and Hansen [11]. Monte Carlo results generally yield excellent agreement with experimental data; however, the results deteriorate statistically at large ranges or narrow observation angles.

Details of a program for generation and curve fitting of Monte Carlo data are contained in Junge [6]. This thesis uses the groundwork of that report to extend study into regions other than ultraviolet light in a homogeneous space.

H. THEORETICAL RESULTS

The purpose of this section is to summarize the effects scattering has on light propagation through clouds as predicted by the theories of the previous sections. In doing this the definitions of effects investigated are given. Actual quantitative relations used in this thesis are reserved for later sections and qualitative descriptions are found here.

Previous Monte Carlo, analytic, and numeric studies of the multiple scattering problem have found that light pulses are distorted in the following manner:

1. Angular Spreading

Angular spreading constitutes an effective decollimation of the incident radiation. It contributes to beam spread, dispersion in angle-of-arrival and loss of spatial coherence.

2. Spatial Spreading

Spatial spreading indicates the dimensional increase of the beam's finite cross section. It is related to the above parameter in that angular spread inside the medium produces spatial spread as the pulse propagates on. Spatial coherence and beam spread are related to this parameter.

3. Multipath Time Spreading

Different distances along various possible propagation paths imply different transit times for a photon. Thus, a short optical pulse will incur pulse broadening after traversing a multiple scattering region. This pulse broadening is called multipath time spreading. The maximum pulse frequency that can be used in communicating is determined by this parameter.

Multipath time spread is often defined as the difference between the average transit time incurred from multiple scattering and the normal transit time in the absence of scattering. This definition is normally used when the time dependence of the input is that of a delta function. The

previous description of multipath time spread concerns input pulses of finite width.

4. Total Transmission

Total transmission describes the amount of irradiance left in the pulse after traversing the medium. It is inversely related to the beam attenuation.

Bucher [1] found that the amount and distribution of multipath time spreading was essentially independent of the detailed shape of the scattering function for sufficiently thick clouds. He also observed that the propagation parameters for sufficiently thin clouds were dependent both on the cloud parameters and on the scattering function.

Gordon [9] found that within certain ranges and angles of practical interest, the flux and beam spread function can be adequately approximated by closed form expressions. His work was related to scattering under water but can be applied equally well to atmospheric scattering when the phase function is very forward peaked.

Time spreads on the order of microseconds to milliseconds have been reported by Bucher [1], Stotts [10,26] and Ishimura and Hong [12]. Danielson, Moore and van de Hulst [7] and Hansen [11] found that the Henyey-Greenstein function adequately approximated the true cloud or haze phase function in determining the reflection and transmission characteristics of clouds.

It is the purpose of this thesis to further verify results found previously and to quantify some of the more

important cloud characteristics with respect to time and spatial spread.

III. STATEMENT OF THE PROBLEMS

A. INTRODUCTION

There are many areas in the study of multiply scattered light that could easily be subject to further study. It was necessary to select a few problems which were assailable using available groundwork and techniques.

B. THE PROBLEMS

The Monte Carlo routine of Ref. 6 was employed to investigate the following problems:

1. Dependence of Spatial Spread on Phase Function

Characterize the spatial spread of light traversing scattering media of various types. Select phase functions of high, moderate and low forward peakedness all with the same $\langle \cos\theta \rangle$ and the same root mean squared scatter angle. Determine the effect on spatial character of light of using the different phase functions. Determine the conditions concerning dependence on phase function.

2. Dependence of Time Spread on Phase Function

Characterize the time spread of light traversing scattering media of various types. Use the same phase functions selected above to determine the effect on temporal character of light of using the different phase functions. Determine the conditions concerning dependence on phase function.

3. Transition Region from Forward to Multiple Scattering

Using the results of the previous two characterizations, determine information concerning the transition from the region of primarily forward scatter to the diffusion region where scatter is directed in all directions. Estimate the accuracy of the information and verify its agreement with other theories.

4. Spatial Characterization of Light Transversing Finite Clouds

Characterize the spatial spread of light traversing a homogeneous medium. Characterize the spatial spread of light traversing the same medium except that a dense cloud of scatterers of finite thickness has been added. Compare the two characterizations and discuss the results.

5. Model the Above Problems

Create a model to simulate the geometries of the above problems and verify at each step that the model is indeed generating results consistent with existing theory.

IV. METHODS OF SIMULATION

A. MONTE CARLO METHOD

The computer routine of Ref. 6 was available for use at the outset of this endeavor. The routine could calculate both spatial and temporal characteristics of light traversing a scattering medium. It required as input the scattering parameters and phase function parameters of the scattering particles of the medium. The routine was restricted to homogeneous atmospheres utilizing a Henyey-Greenstein phase function or a Modified Henyey-Greenstein phase function [16], and a given portion of Rayleigh scattering. It was necessary to modify the routine so that it adequately represented the geometry of a finite cloud and so that any arbitrary phase function could be adapted to it. These changes are summarized in the following sub-sections.

1. The Model

Figure 4 represents the model used in this simulation. The model is identical to that of Ref. 6 with respect to photon path generation and accountability. The figure depicts a cloud on whose left boundary the photons are incident at the center of the accountability shells, and at whose right boundary time and spatial information is tabulated. Figure 5 expands the cloud and lists the names of the parameters used in the simulation. KSCA1 and KSCA2 represent the scattering coefficients in inverse kilometers of the inside and outside

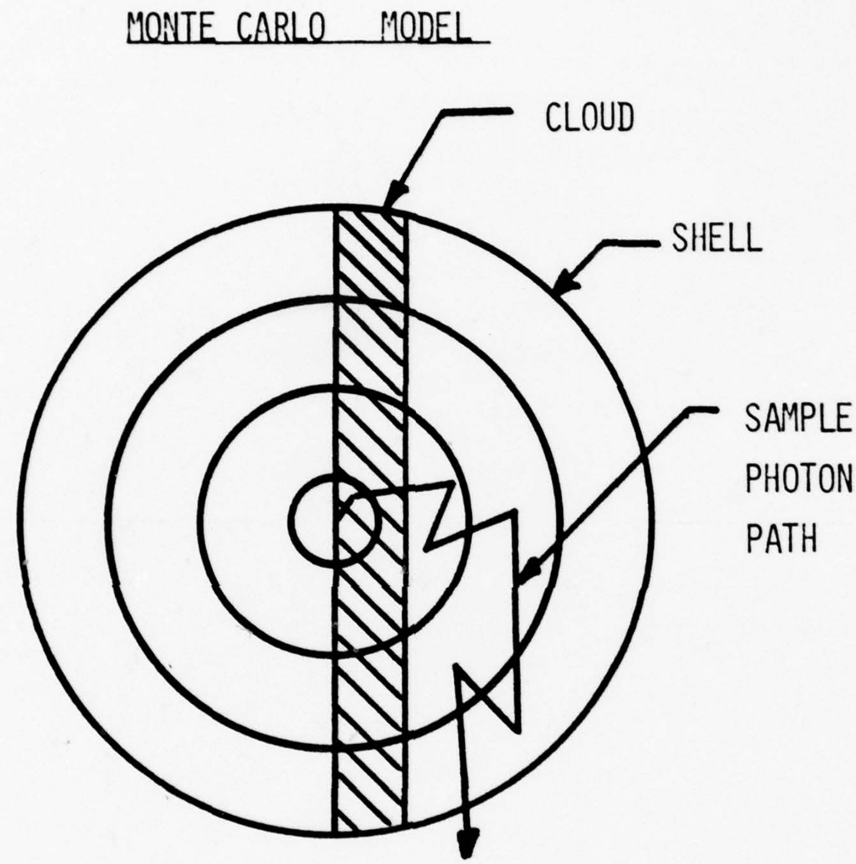
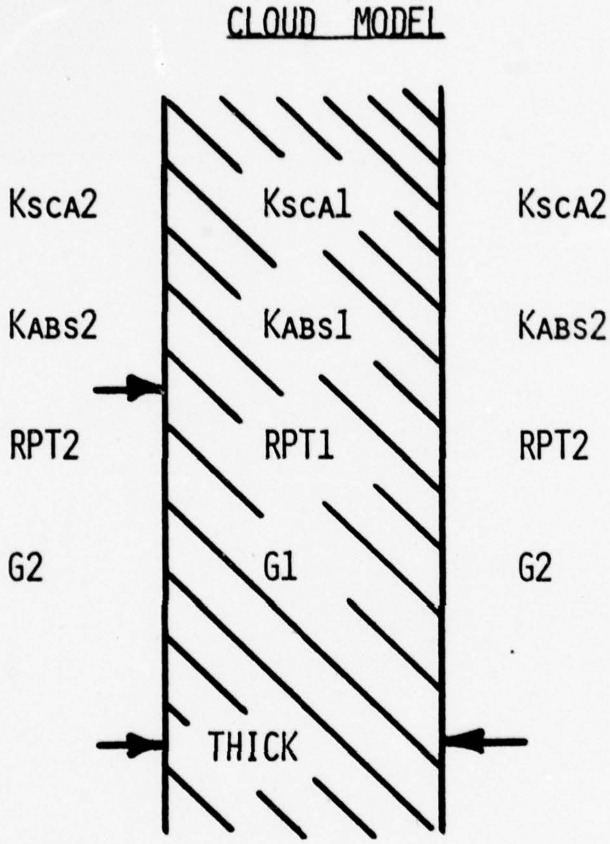


Figure 4

Monte Carlo Model



WHEN PHASE FUNCTION OTHER THAN HENYEY-GREENSTEIN IS USED, IT IS NOTED ON THE RESPECTIVE GRAPH

Figure 5

Cloud Model

media respectively. Likewise, KABS1 and KABS2 represent the absorption coefficients of the two media. RPT1 and RPT2 represent the ratio of particulate to total scatter and G1 and G2 represent the Henyey-Greenstein phase function parameters when used in this simulation. THICK is the physical thickness of the cloud in kilometers. Appendix C gives details on the actual modification of the routine.

At each crossing of a photon from inside of the cloud to beyond the cloud, the total distance traveled is determined and tabulated in time of arrival bins. Time spread and delay information is presented using these bins. A running summation and counter are used to calculate the mean and standard deviation of time required to reach the cloud boundary.

One other change from the original routine is conversion of dimensions from units of extinction lengths to units of kilometers. Only minor changes in program logic were required to make this change.

Typical values of atmospheric parameters were used in most simulations. A summary of the parameters used is given in Table I. All diagrams of spatial and temporal character include parameters used in the specific case.

The Monte Carlo results of Bucher [1] were very useful as guidelines for determining correct operation of the model created. Comparisons of the two models appear later in this section.

TABLE I

Scattering Coefficients used in the Simulation

	KSCA(km^{-1})	Albedo
Clear Atmosphere	.391	1.0
Light Haze	.90	1.0
NOSC Fog	6.37	1.0
Water Cloud C.2	11.18	1.0

2. Phase Functions

As mentioned in the statement of the problem, three phase functions of different peakedness yet equal in $\langle \cos\theta \rangle$ were needed to investigate the problem. The following phase functions were employed:

a. Henyey-Greenstein Phase Function

Figure 1 depicts the functional dependence of scatter on angle off direction of incidence, θ. The closed form expression for the Henyey-Greenstein function has been stated earlier in Equation 8. A G value of .83 was selected so as to match other phase functions used in the simulation. The peak of the phase function is only five units. Reference 6 explains its use in the Monte Carlo routine.

b. Numerical Data Input for Arbitrary Phase Functions

The other two phase functions required development of a method for inputting phase functions consisting only of data pairs and no closed functional form. They are that representing Water Cloud C.2 [21] at .53 microns and that of

NOSC Fog. The first was generated by use of the program of Appendix B, the second by a similar calculation using an experimentally obtained water particle distribution. Appendix A describes the details of generating a random scatter angle weighted by these arbitrary phase functions defined by data pairs.

c. Rayleigh Phase Function

This routine can also simulate a fractional amount of the total scatter as Rayleigh scatter. In this work the use of this capacity was minimal because in most cases water clouds of predominately large particles were simulated.

3. <u>Automation of Contour Plotting</u>

Contours of equal relative photon flux, as described in Ref. 6, were used in this work for spatial characterization. A method of machine computation was adapted to the DRLITE routine. This allowed the computer to perform the tedious interpolation made before by pocket calculator. Although useful in some instances, the procedure proved quite computer time consumptive. Its usefulness was very pronounced in comparing the Monte Carlo routine with Gordon [9] as will be discussed later. Appendix D gives details of the adaption of this method to the routine.

B. ANALYTICAL METHODS

As explained in the previous section on theoretical preliminaries, analytical methods ordinarily lead to solutions which are applicable only under certain conditions. This is so

because in order to reach a usable solution, approximations have to be made. Nevertheless, in the regions of their applicability, analytical solutions are excellent sources of the information needed to verify results generated by other more universally applicable methods. In this work two analytical treatments were employed for just that reason. Gordon's [9] paper on practical approaches to underwater scattering was used to verify the spatial output of the Monte Carlo routine created and Stott's [26] closed form expression for time spread was used to verify the temporal output of the program. The two treatments are summarized in the following paragraphs.

1. Effective Attenuation Coefficient (EAC) Method

Gordon's paper [9] presents the concept of an effective attenuation coefficient which considerably simplifies underwater multiple scattering calculations. Concise expressions for the total flux through an aperture and the beam-spread function are defined in terms of the EAC. The derivation of these expressions is included for easy reference as Appendix E as is a documentation and description of the program adapting them to machine computation. It is interesting to note that Gordon's treatment is actually a truncated version of the exact analytical solution of the multiple scattering problem using the small angle approximation provided by him [28]. The truncation, in effect, considers the scattering phase function as a forward scattering delta function, thus does not consider higher order terms of the exact solution. It is not surprising then that the result is only applicable under certain conditions.

Fortunately these conditions are found often in the underwater optical world. The comparison of this method to the Monte Carlo method is found in the next section. It should be kept in mind that Gordon's treatment, even in the truncated manner, closely agrees with experimental data [29]. The reader is referred to Appendix E for further details on the EAC method.

2. Closed Form Time Spread Expression

Stott's treatment [26] of time spread in a multiple scattering region is short, concise and seems to agree well with existing time spread data. It was therefore chosen as an additional means for verification of time spread results given by the Monte Carlo method. The details in deriving Stott's closed form solution for time spread are located in Appendix G for easy reference. The result is the average additional multipath time required for a photon to traverse a scattering medium with physical thickness z, optical thickness τ, albedo ω_o and RMS scatter angle γ_o. The time spread is

$$L = \frac{z}{c}\left(\frac{.30}{\omega_o \tau \gamma_o^{\,2}}\left[(1+\frac{9}{4}\,\omega_o \tau \gamma_o^{\,2})^{1.5}-1\right]-1\right) \qquad (12)$$

This result is compared to Monte Carlo results in later sections.

C. COMPARISON OF METHODS EMPLOYED

A problem usually encountered in mathematically modeling any physical situation is to ensure at all times that the model

employed is turning out reasonable answers. To verify that this was in fact the case, frequent cross-checks were made between the models. The proof of the theory follows, of course, in agreement between the models and/or agreement with experimental observations. This section compares the results of the Monte Carlo model with the analytical models as well as with the results of Bucher's [1] independently created Monte Carlo model. Once confidence is established, conclusions can be drawn about the regions of effectiveness of each model. Other more routine checks for error are noted in Appendix F.

1. Monte Carlo vs. Analytical

a. Spatial Characterization

The EAC method derived and verified in Appendix E is compared to the Monte Carlo method in this section. First, the method of spatial characterization adapted from Ref. 6 is briefly explained.

(1) Description of Relative Flux Contours. At each angular bin of each reference shell the relative flux is given by

$$\text{Relative Flux} = \frac{\text{\# Photons Passing Through Bin}}{\text{Total Number Ejected x Area of Bin}} \qquad (13)$$

The negative logarithm of this quantity is calculated for each bin. This array of values is interpolated as described in Appendix D, and contours of equal relative photon flux are drawn. Likewise, the negative logarithm of the beam spread

function evaluated by the EAC method is evaluated at a similar array of spatial positions and equal flux contours are drawn. The two sets of contour lines are compared in the following paragraphs.

(2) Comparison. Flux contours calculated using Henyey-Greenstein G parameters of .70 and .95 for each albedo of .80 and .90 were drawn at integer values of the negative logarithm. Figures 6 through 9 show these comparisons. The dotted flux contours are those of the EAC method from zero to 45 degrees and the solid contours are those of the Monte Carlo method. The term S on the graph is the scattering coefficient as well as the albedo since the extinction coefficient is unity in each case. Dimensions are in kilometers but in this case one kilometer is equivalent to one extinction length.

In general, agreement is good between the methods for highly peaked, G = .95, phase functions at distances less than 10 extinction lengths and angles varying from one to 20 degrees. This comparison agrees with results given in Gordon [9]. It should be noted that angular agreement is nearly independent of albedo whereas range agreement is better when large absorption is present.

(3) Conclusions. Inside of a few degrees at moderate to large extinction lengths the EAC method becomes undependable in most cases. The Monte Carlo and EAC methods agree well in regions where agreement should be expected. The Monte Carlo method has the advantage over the EAC method when

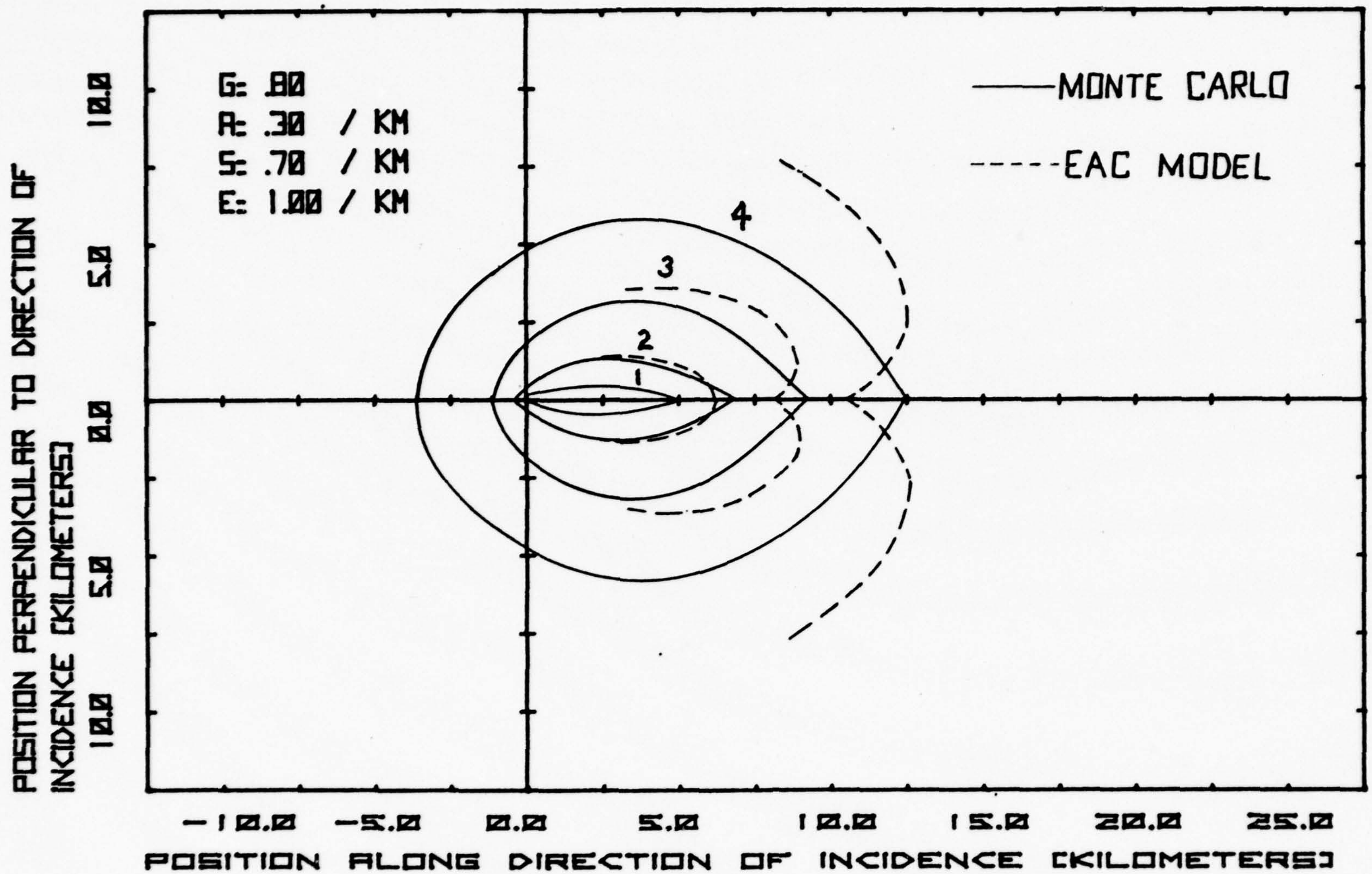


Figure 6

EAC Model vs. Monte Carlo Model

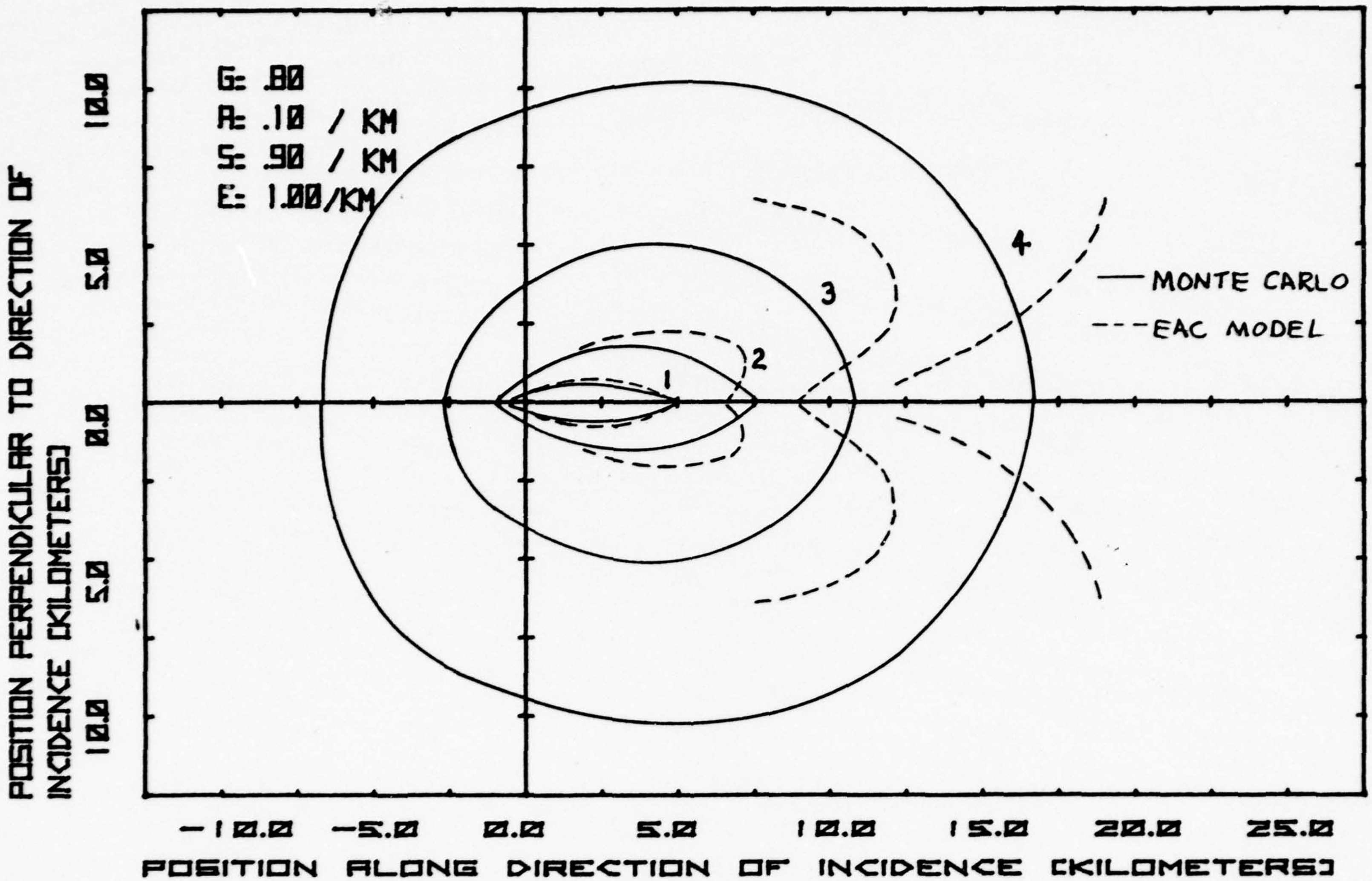


Figure 7

EAC Model vs. Monte Carlo Model

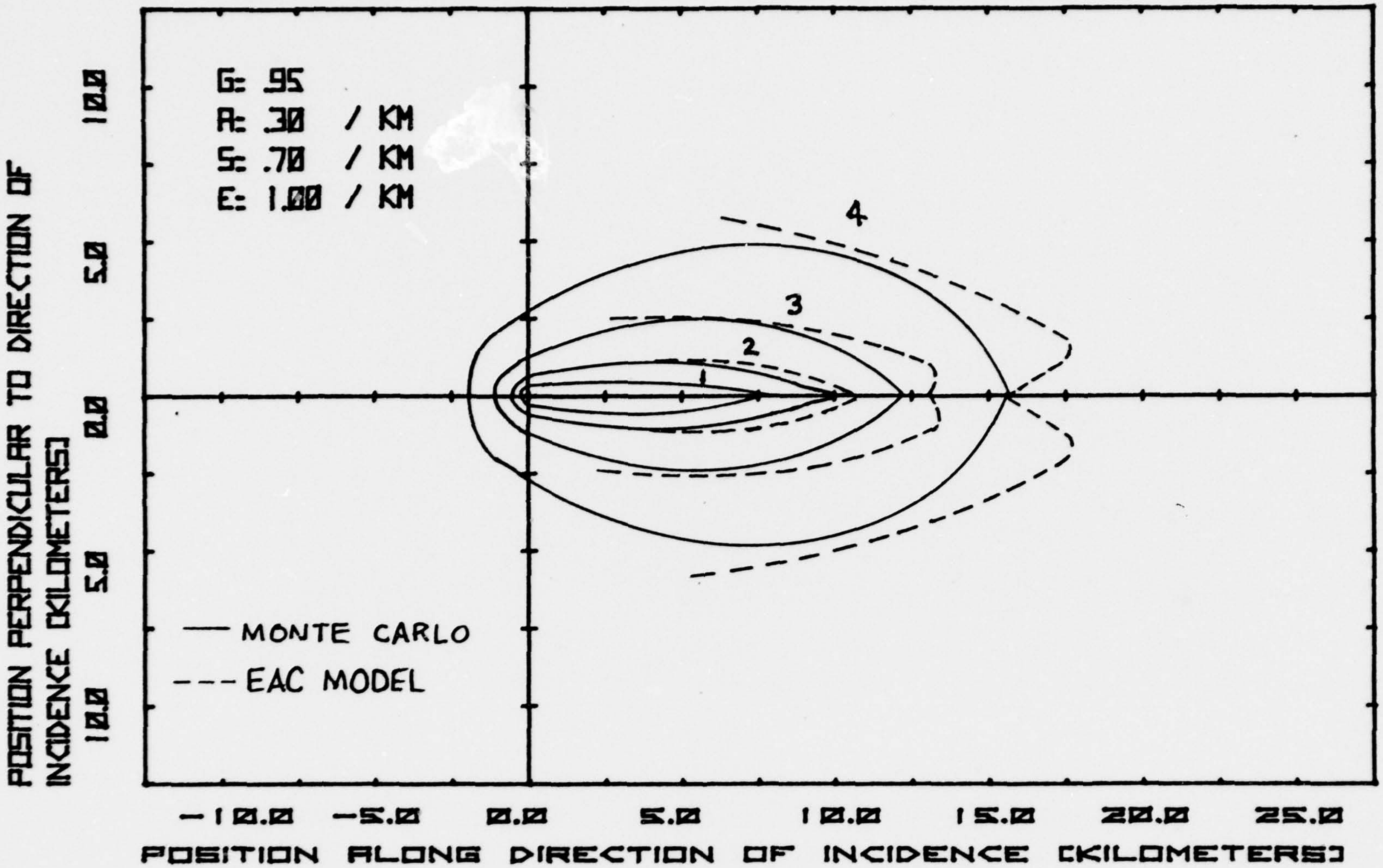


Figure 8

EAC Model vs. Monte Carlo Model

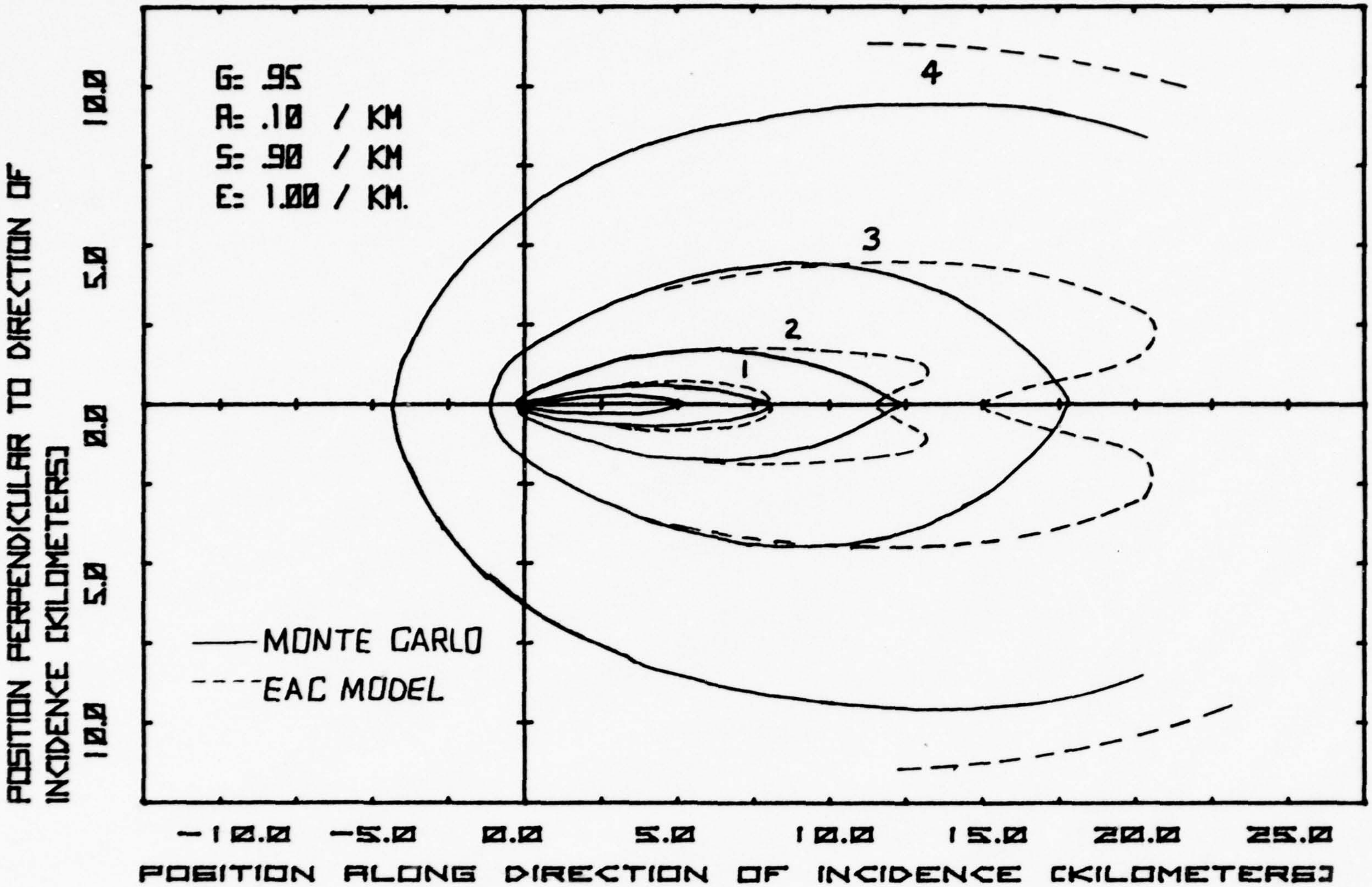


Figure 9

EAC Model vs. Monte Carlo Model

considering atmospheric scattering because in this case absorption is small and large optical thicknesses are encountered often. Confidence can be placed in the Monte Carlo routine for spatial characterization of light in a multiple scattering region.

b. Temporal Characterization

Stotts's [26] closed form expression for time spread which is summarized in Appendix G is compared to time spread found using the Monte Carlo method in this section. The technique used in tabulating time spread information in the Monte Carlo model was explained earlier in the section titled "The Model".

(1) Comparison. Multipath time spread is defined as the difference between the average transit time incurred from multiple scattering and the normal transit time in the absence of scattering. The expression for time spread derived by Stotts is given in Equation 12. This equation is plotted in Figures 10 through 14 with an albedo of one, an RMS scatter angle of .65 and optical thickness, τ, ranging from zero to 100 for physical thickness values, z, of .5, 1.0, 2.0, 3.0 and 4.0 kilometers. Monte Carlo data points are marked with X's using various τ values with other parameters the same. Also plotted are Bucher's [1] data which is explained later. In general, agreement is within one microsecond for optical thicknesses less than 10 and within five microseconds for optical thicknesses between 10 and 60.

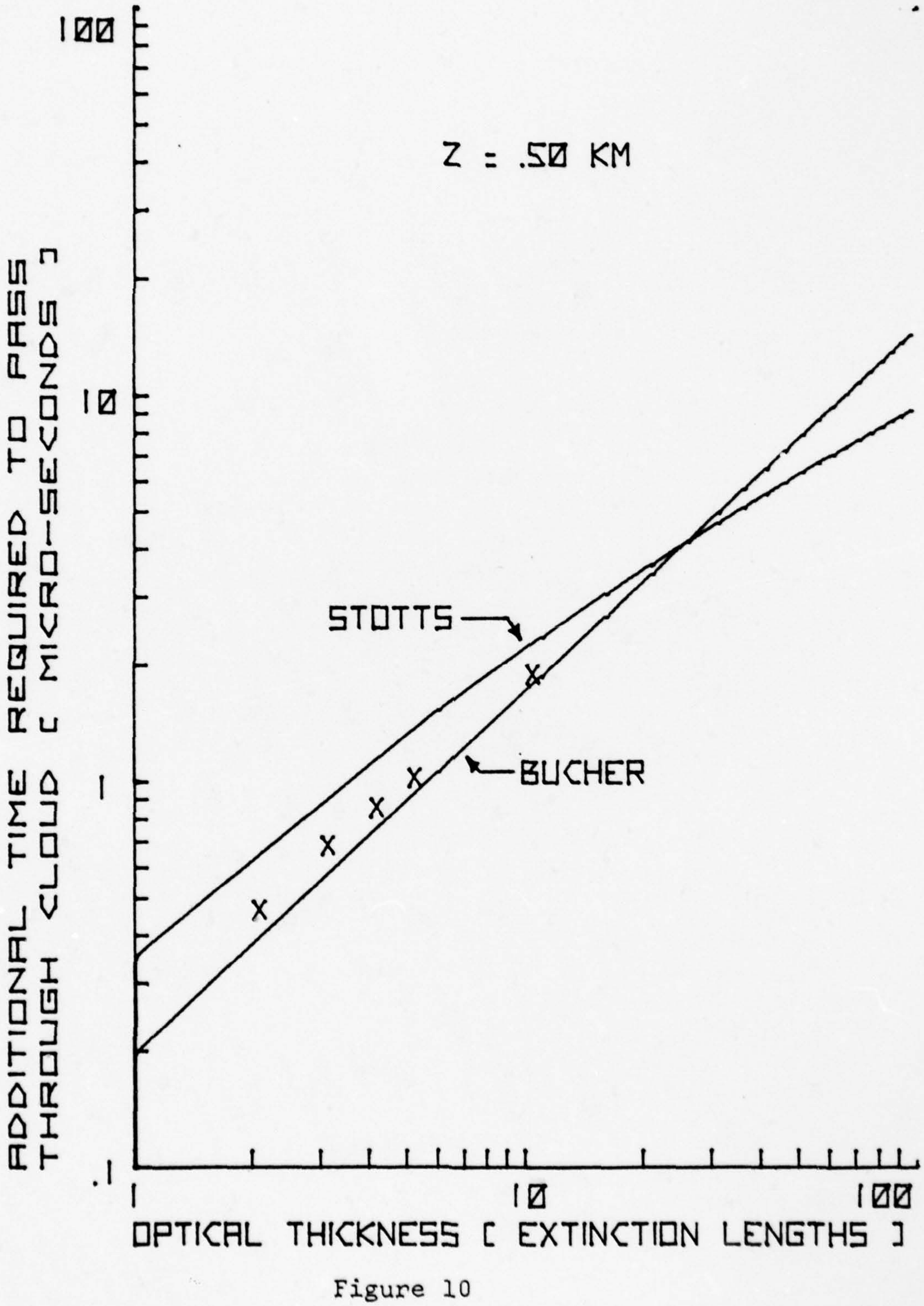


Figure 10

Three Model Time Spread Comparison

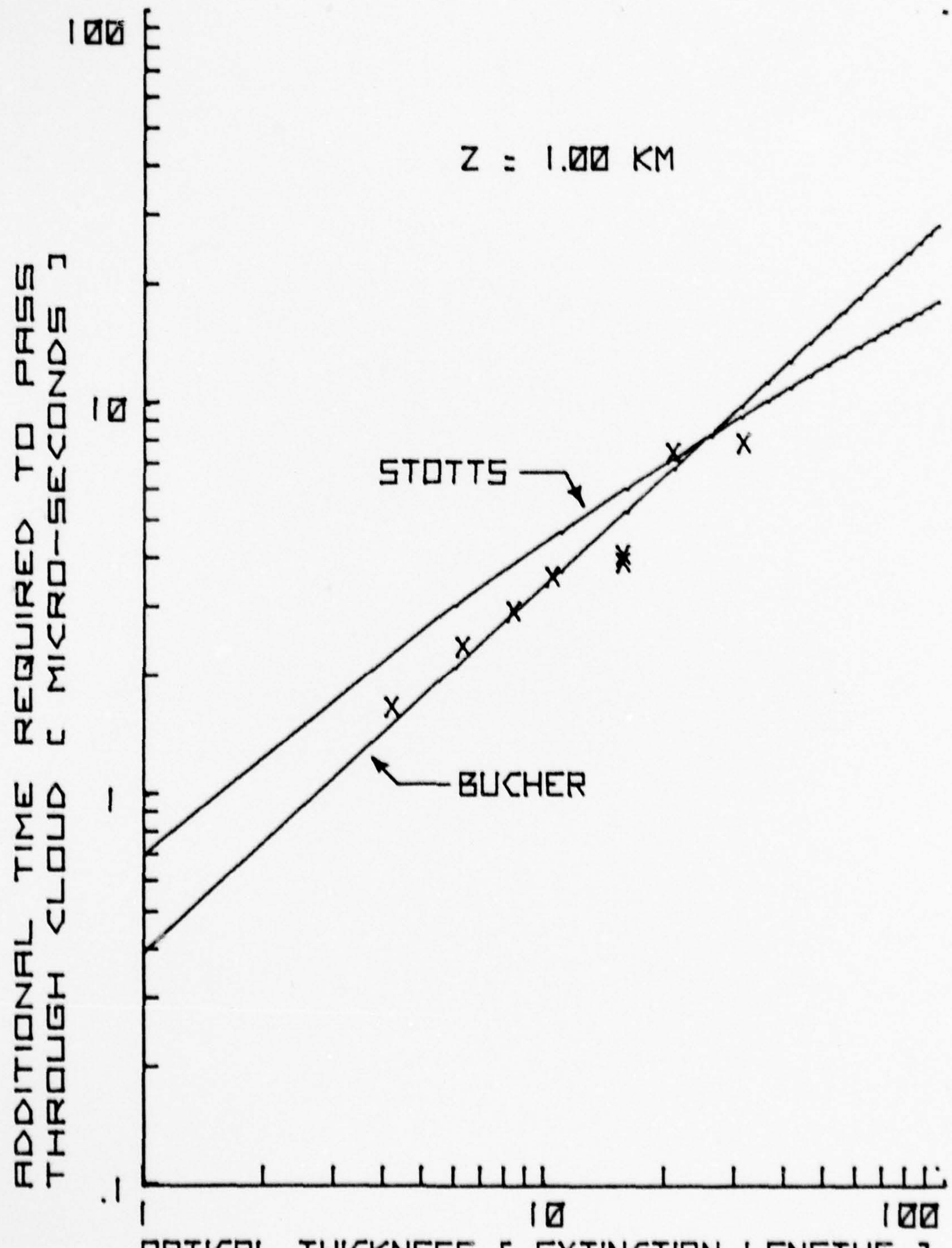


Figure 11

Three Model Time Spread Comparison

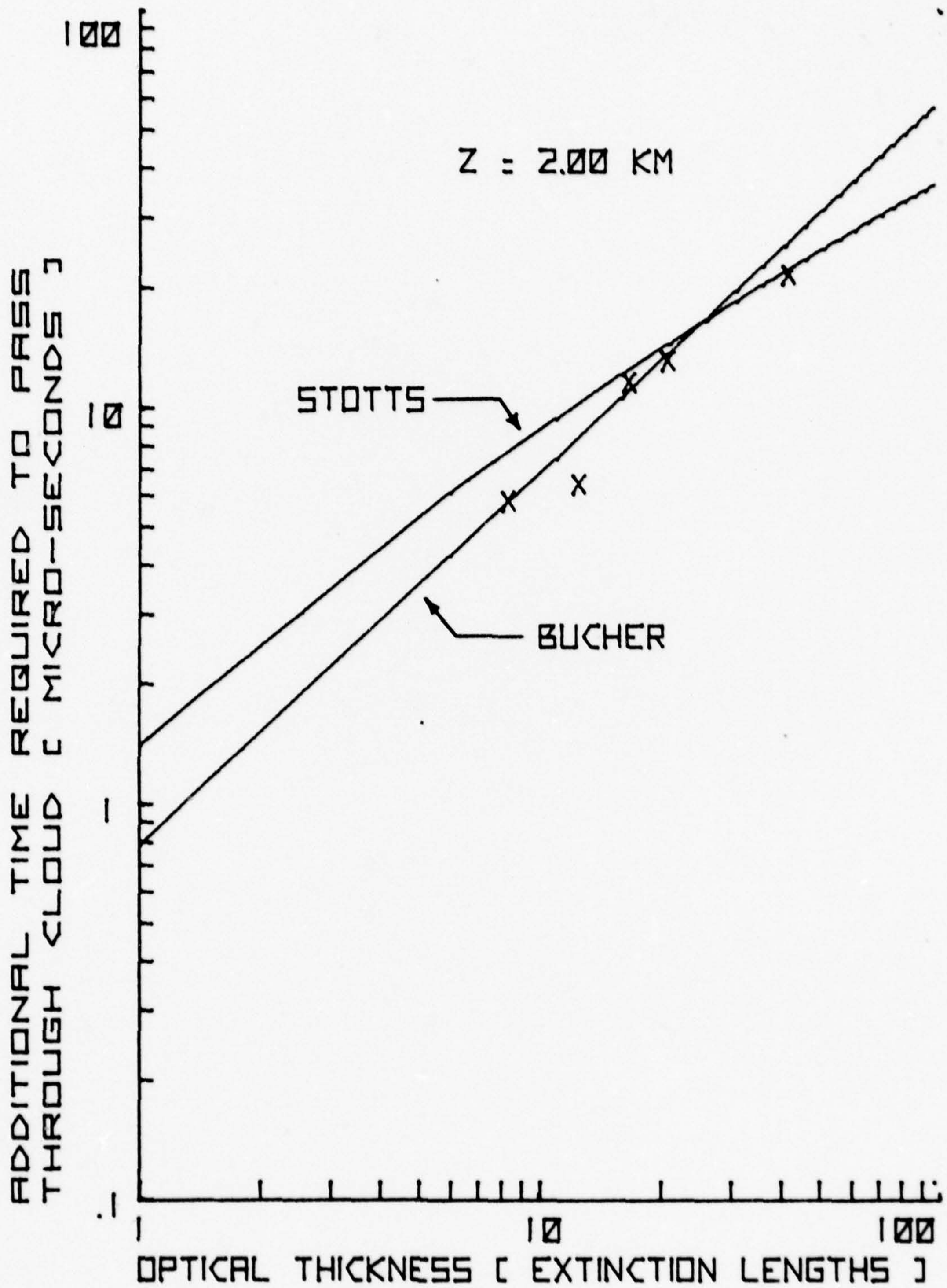


Figure 12

Three Model Time Spread Comparison

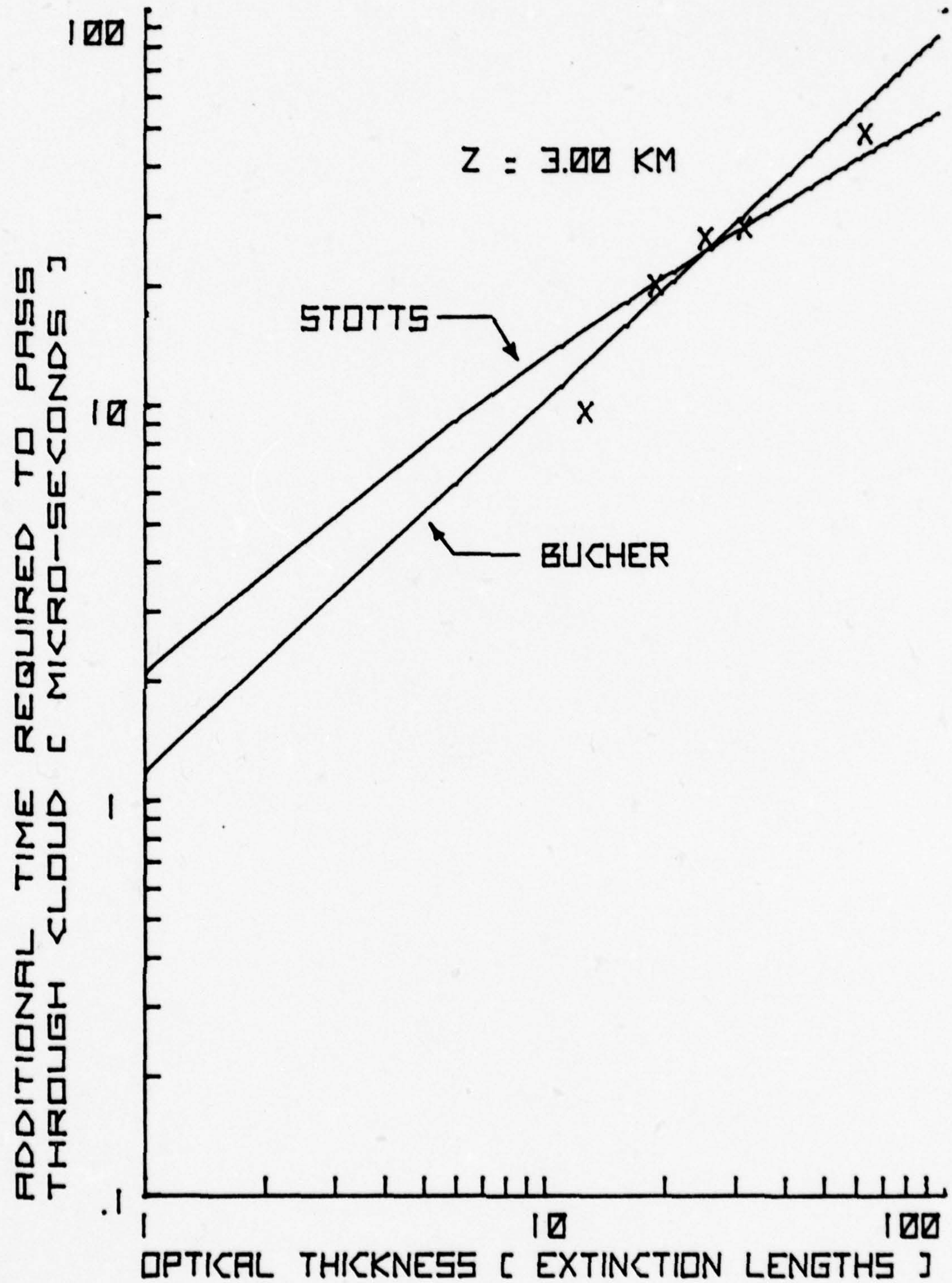


Figure 13

Three Model Time Spread Comparison

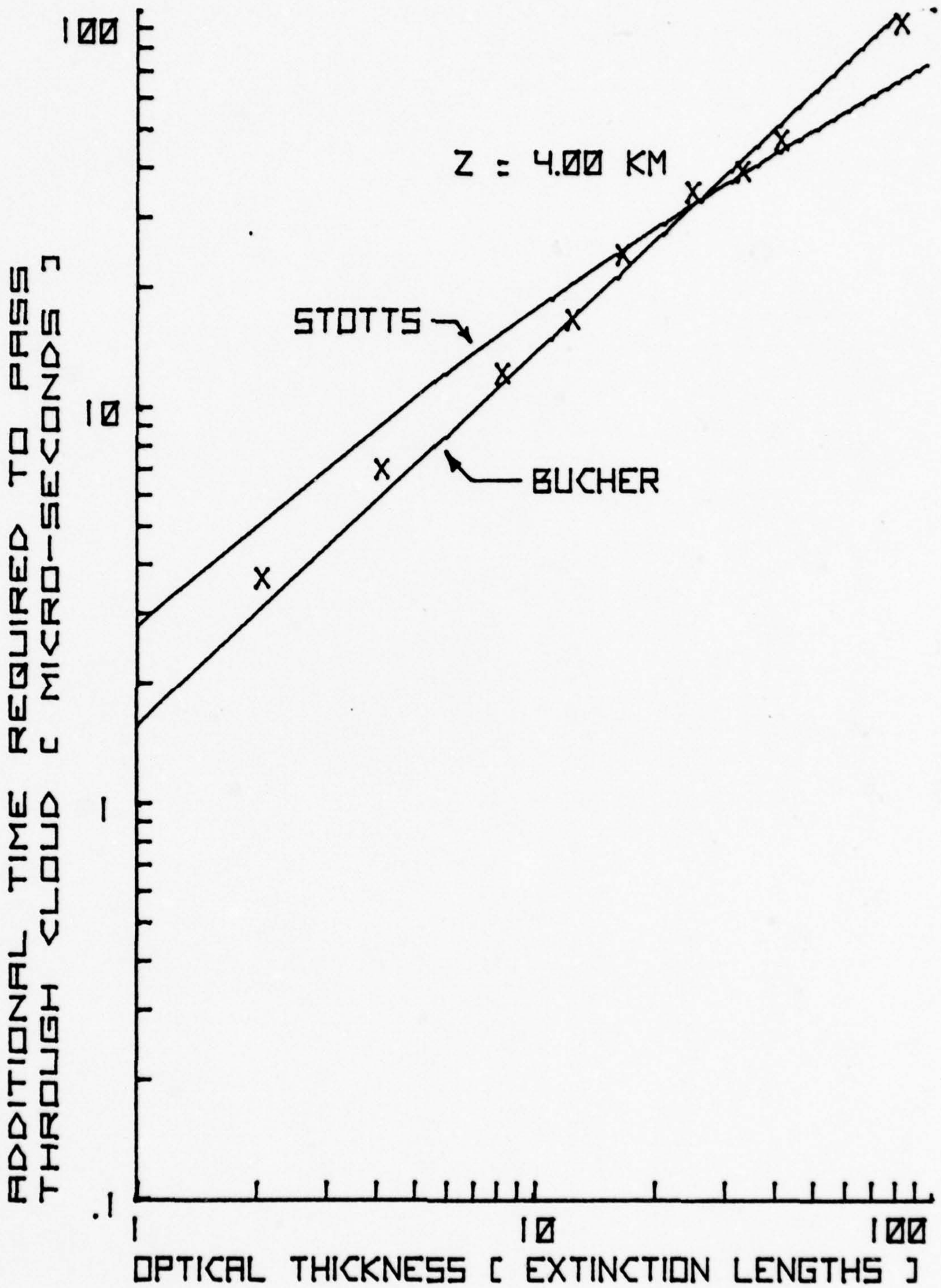


Figure 14

Three Model Time Spread Comparison

(2) Conclusions. The Monte Carlo and closed form methods agree well in regions where agreement should be expected. Because of this agreement, confidence can be placed in the Monte Carlo routine for temporal characterization. The Monte Carlo routine has the advantage over the closed form expression in that different phase functions can be simulated, thus allowing study of the effects of the details of the phase functions on time spread of light in a scattering medium. Because time spread is expressed quite well by the closed form expression, little dependence on these details is expected. The dependence is discussed in later sections.

2. This Monte Carlo Model vs. Bucher's Monte Carlo Model [1]

a. Spatial Characterization

The Monte Carlo method is tested for agreement with Bucher's [1] Monte Carlo results. Because both methods are built with nearly the same geometry, nearly identical results are expected. First the graphic representation of data used by Bucher needs explanation.

(1) Description of Scaling Method. Bucher [1] defines for a scattering medium the diffusion distance

$$D_d = \frac{D}{1-\langle\cos\theta\rangle} , \qquad (14)$$

where D is the mean free path between collisions and $\langle\cos\theta\rangle$ is the average cosine of the scattering angle as before. Using this formula, the effective scattering thickness τ_d is given by the relation

$$\tau_d = \frac{T}{D_d} = \tau(1-\langle\cos\theta\rangle) \qquad (15)$$

where T is the physical thickness of the cloud and τ is the usual optical thickness. The spatial results are then presented by graphically displaying the average displacement $\langle r\rangle$ in relation to the diffusion thickness, D_d, as a function of effective scattering thickness τ_d. Also presented is the critical transverse displacement, r_c, in relation to D_d inside which 50 per cent of the photons exit the cloud at effective scattering thickness, τ_d. The reader is referred to Bucher [1] for further details.

(2) Comparison. Monte Carlo calculations were made and interpreted in terms of the units described above. Figures 15 and 16 show the results at various effective scattering thicknesses using an albedo of unity and a Henyey-Greenstein function for which G = .83. The X's represent data collected here and the continuous line is a best fit curve to Bucher's data using the same parameters. As can be seen, agreement is excellent in both cases.

(3) Conclusion. Spatial characterization by this Monte Carlo routine is consistent with a previously developed routine. This creates confidence in the routine. It can now be used to study the effect phase function details have on the spatial character of light.

b. Temporal Characterization

The time spread data of the Monte Carlo routine is compared to Bucher [1] in this section. Two methods of

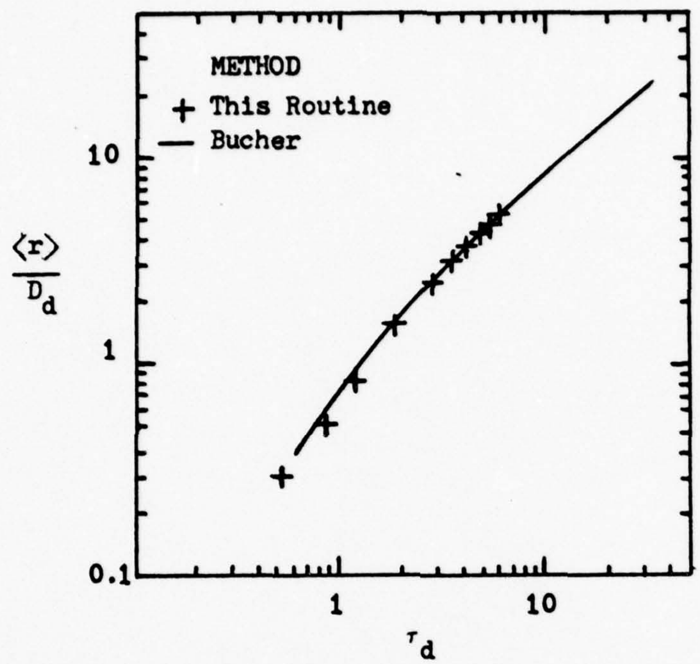


Figure 15

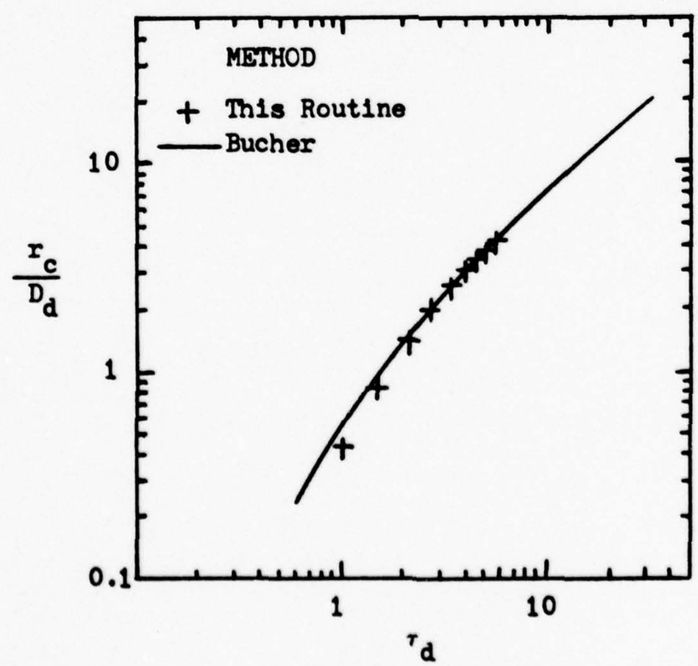


Figure 16

This Monte Carlo vs. Bucher's
Monte Carlo [1] Spatial Spread Comparison

presentation are used. One uses time bins, the other uses Bucher's best fit equation for time spread.

(1) Comparison. Bucher's equation for multipath time spread as defined in an earlier section is,

$$L(\tau_d) = \frac{.62T}{c}(\tau_d)^{.94} = 3.91 \times 10^{-10} T(\tau_d)^{.94} \quad (16)$$

where $L(\tau_d)$ is the time spread in seconds, T is the physical thickness of the cloud in meters, τ_d is the effective scattering thickness of the cloud and c is the speed of light in meters per second. This function is plotted alongside Monte Carlo data and Stott's closed form equation in Figures 10 through 14 for an albedo of unity, a Henyey-Greenstein phase function with G = .83, and a physical thickness of .5, 1.0, 2.0, 3.0 and 4.0 kilometers.

Agreement is well within one microsecond for nearly the entire range which is well within statistical error. In these trials only 50 - 500 photon histories were tabulated which seems to be an indication that collection of time spread information does not require unwieldy amounts of computer time.

Another comparison of time spread information is graphically displayed in Figures 17 and 18. The number of photons arriving in each time bin is normalized to the maximum number in any bin and plotted as a function of time in microseconds. Bucher's data is superimposed by a dotted line on the time bin data presented. Figure 17 plots results for an optical thickness of 15 and Figure 18 for an optical thickness of 30. Other parameters are listed on the graphs. Agreement once again is very good between the methods.

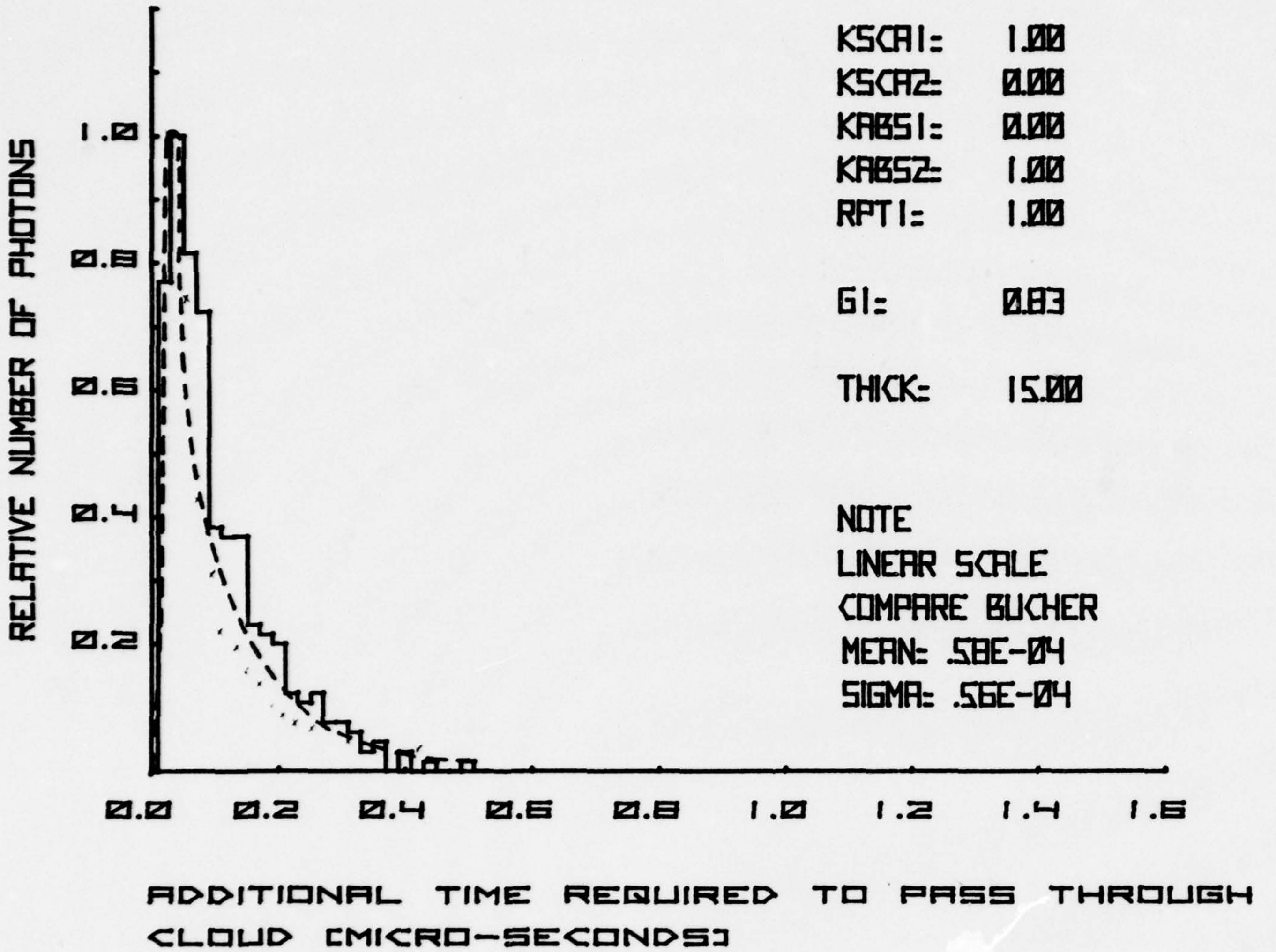


Figure 17

This Monte Carlo vs. Bucher's Monte Carlo [1] Time Spread Comparison

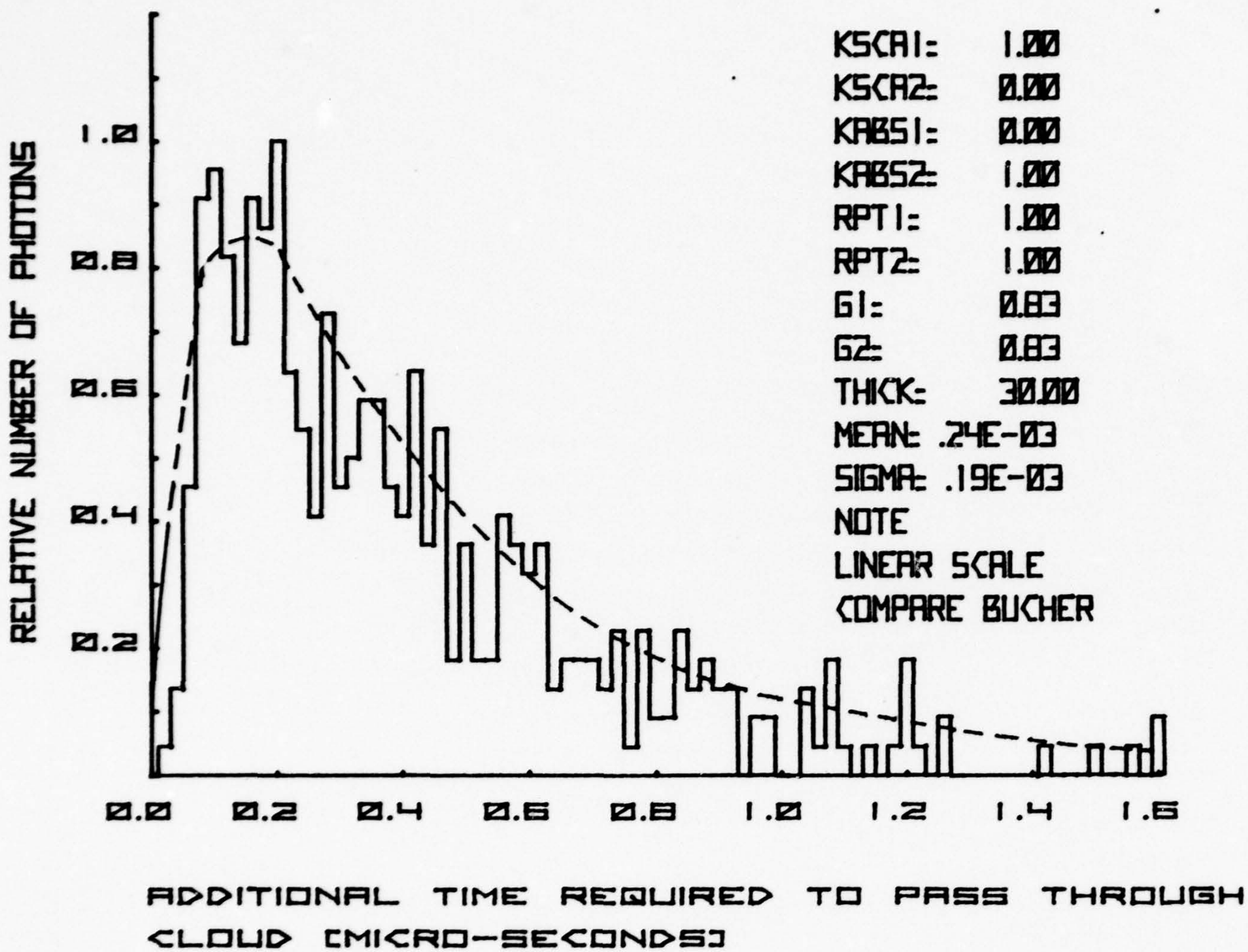


Figure 18

This Monte Carlo vs. Bucher's Monte Carlo [1] Time Spread Comparison

(2) Conclusion. Once again, confidence is ensured in temporal characterization of time spread and pulse spread by the Monte Carlo routine. The routine can now be used in investigating time spread dependence on phase funtion.

V. RESULTS

A. SPATIAL SPREAD DEPENDENCE ON PHASE FUNCTION

In this section five graphs are used to display the dependence of the spatial spread of light on phase function when traversing a scattering medium. All five diagrams show the relative flux contours described earlier in this thesis and in Ref. 6. Each figure lists the parameters and the phase function used in the simulation. Each axis is in units of kilometers. Incidence of photons is at the origin and to the right in all cases. The left edge of the cloud, if it exists, is along the axis perpendicular to direction of incidence and the right edge of the cloud is indicated by a dotted line if it is within the boundaries of the graph. For easy reference, Table II tabulates the parameters and phase function used in Figures 19 through 36.

The phase function of Water Cloud C.2 is used in Figure 19 to show the peakedness of the relative flux contours out to at least 20 extinction lengths when the albedo is only .90. A "toe" appears on the contours at small angles due to the high degree of forward scattering of Water Cloud C.2.

Figures 20 and 21 compare the flux contours due to NOSC Fog and Henyey-Greenstein using G = .83, respectively. The flux contour for NOSC Fog is more forward peaked than that of the Henyey-Greenstein function for distances less than three kilometers or 19 extinction lengths but flux contours are very

TABLE II

Tabulation of Parameters and Phase Functions used in Figures 19 - 36

	Figure	KSCA1 (km^{-1})	KSCA2 (km^{-1})	KABS1 (km^{-1})	KABS2 (km^{-1})	RPT1	RPT2	G1	G2	THICK (km)	MEAN (μ-secs)	SIGMA (μ-secs)	PHASE F
Spatial	19	.90	.90	.10	.10	1.0	1.0	N/A	N/A	N/A	N/A	N/A	WC C.2
	20	6.37	0.0	0.0	1.0	1.0	N/A	N/A	N/A	00	N/A	N/A	NOSC
	21	6.37	0.0	0.0	1.0	1.0	N/A	.83	.83	00	N/A	N/A	H.G.
	22	11.18	0.0	0.0	1.0	1.0	1.0	N/A	N/A	00	N/A	N/A	WC C.2
	23	11.18	0.0	0.0	1.0	1.0	1.0	.83	.83	00	N/A	N/A	H.G.
Temporal	24	4.0	0.0	0.0	1.0	1.0	N/A	N/A	N/A	1.0	1.25	2.83	NOSC
	25	4.0	0.0	0.0	1.0	1.0	N/A	N/A	N/A	1.0	1.15	2.42	WC C.2
	26	4.0	0.0	0.0	1.0	1.0	N/A	.83	N/A	1.0	1.66	3.07	H.G.
	27	8.0	0.0	0.0	1.0	1.0	N/A	N/A	N/A	1.0	2.92	4.16	NOSC
	28	8.0	0.0	0.0	1.0	1.0	N/A	N/A	N/A	1.0	2.90	4.54	WC C.2
	29	8.0	0.0	0.0	1.0	1.0	N/A	.83	N/A	1.0	2.85	4.19	H.G.
	30	15.0	0.0	0.0	1.0	1.0	N/A	N/A	N/A	1.0	4.33	4.66	NOSC
	31	15.0	0.0	0.0	1.0	1.0	N/A	N/A	N/A	1.0	4.61	5.34	WC C.2
	32	15.0	0.0	0.0	1.0	1.0	N/A	.83	N/A	1.0	4.38	4.41	H.G.
Cloud	33	.10	.10	.01	.01	1.0	1.0	.90	.90	N/A	N/A	N/A	H.G.
	34	3.0	.10	.01	.01	1.0	1.0	.90	.90	3.0	N/A	N/A	H.G.
	35	.39	.39	.01	.01	.10	.10	N/A	N/A	N/A	N/A	N/A	WC C.2
	36	11.18	.39	.01	.01	1.0	.10	N/A	N/A	.50	N/A	N/A	WC C.2

(In Figures 20-23, 35 and 36 the dots represent the data points to which the contours were fit)

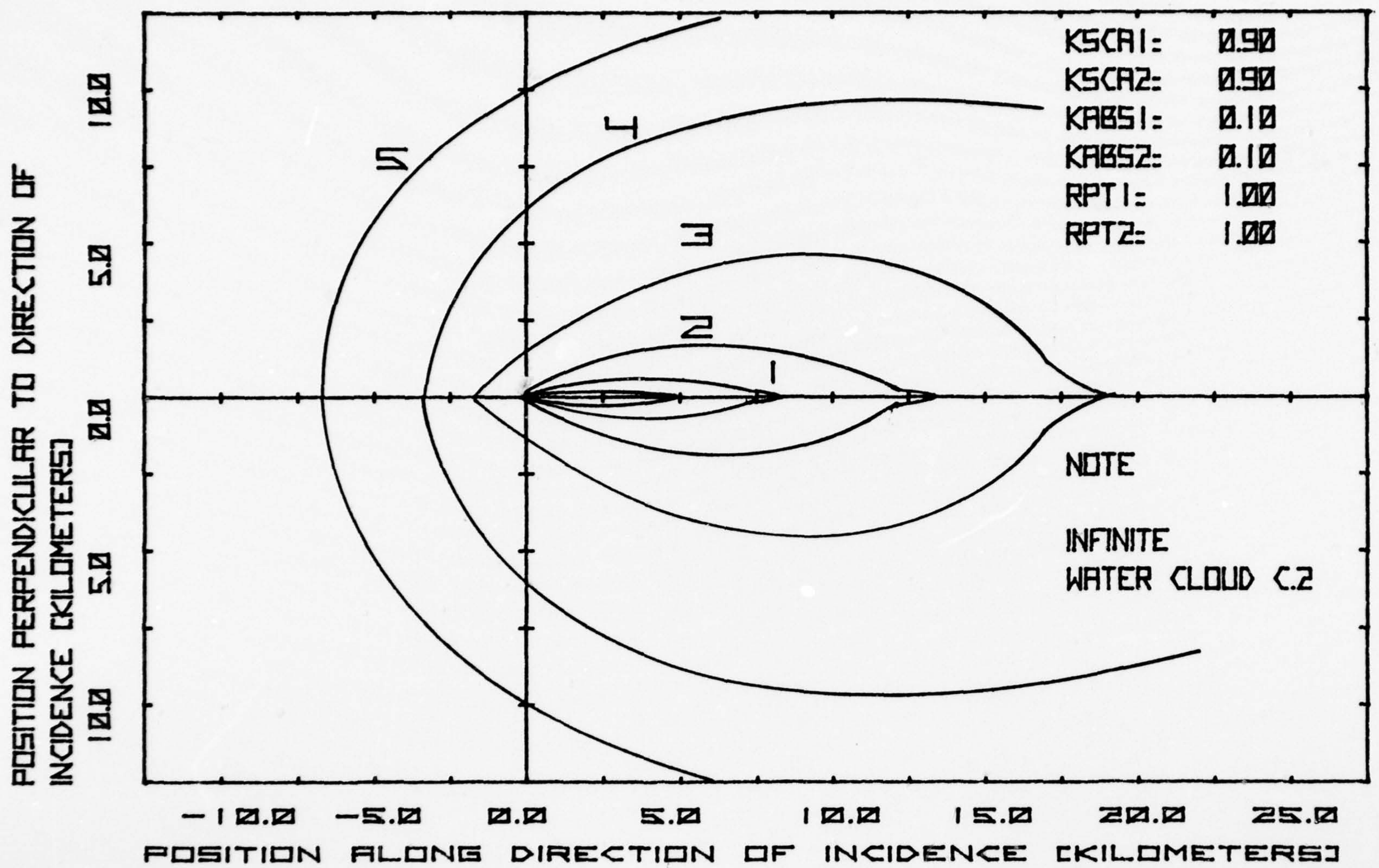


Figure 19

Spatial Spread Dependence on Phase Function

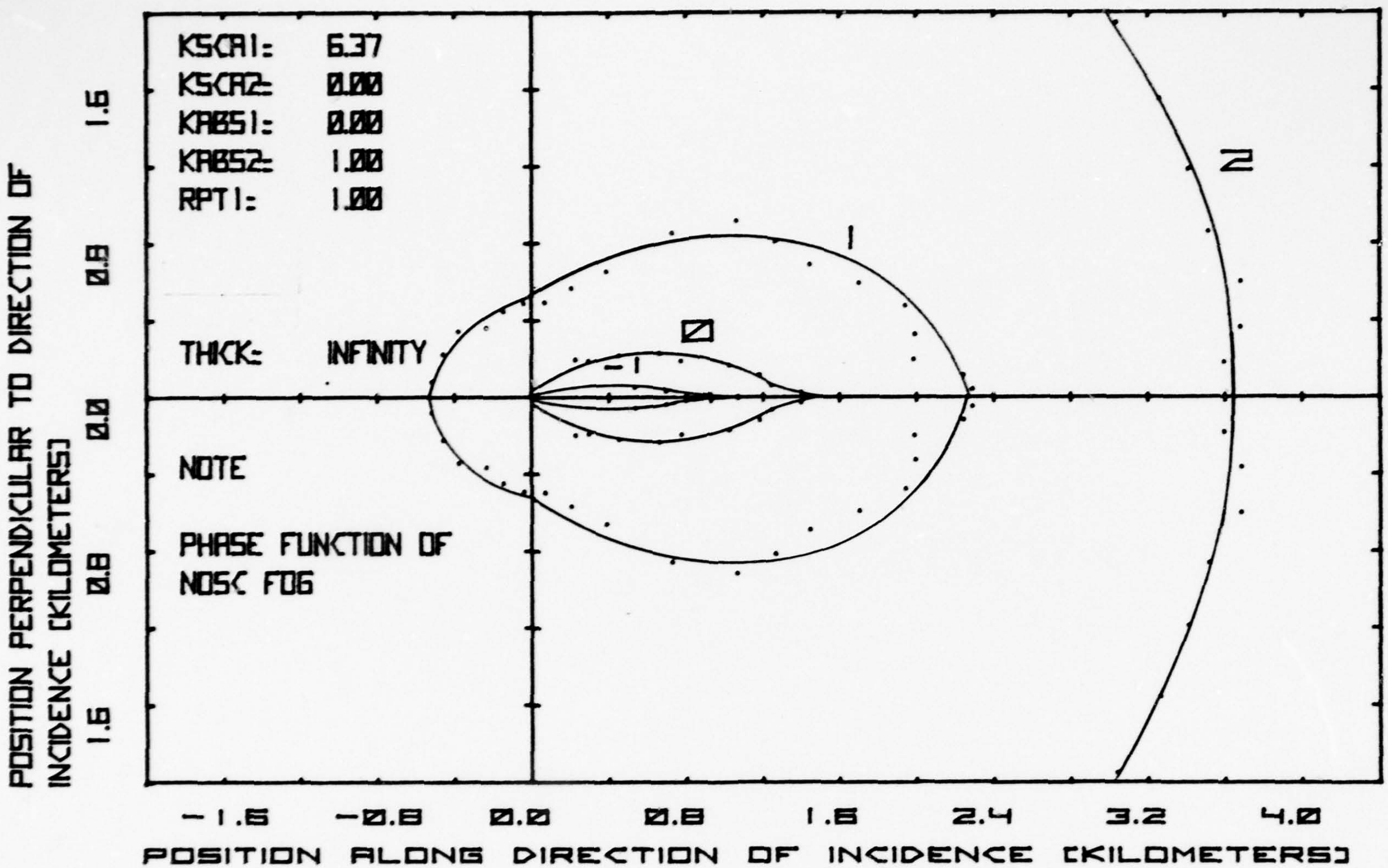


Figure 20

Spatial Spread Dependence on Phase Function

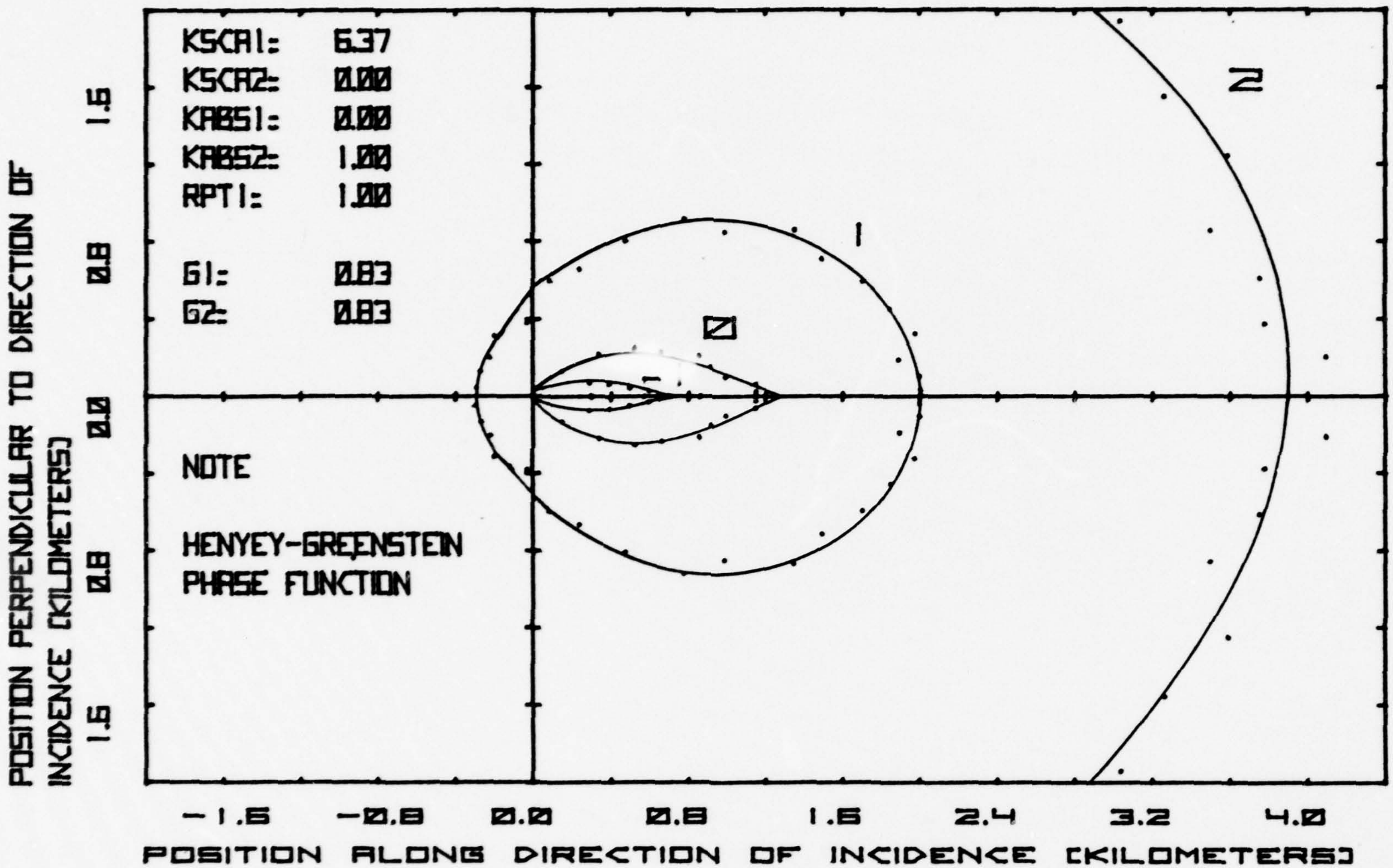


Figure 21

Spatial Spread Dependence on Phase Function

similar at distances greater than three kilometers. Outside of 20 extinction lengths the contours and therefore the spatial character of the light they represent become independent of the phase function used. The contours of these two graphs are much less forward peaked than that of Figure 19 because the albedo in the former is unity.

Figures 22 and 23 show similar results when comparing results when Water Cloud C.2 [21] and Henyey-Greenstein were used, respectively. In all figures the cloud was extended to infinity at the right of the origin, explaining the discontinuity in contour slope.

B. TEMPORAL SPREAD DEPENDENCE ON PHASE FUNCTION

In this section nine histograms and one table are used to display the dependence of time spread of light on phase function when traversing a scattering medium. All nine diagrams list the parameters in the same fashion as in the last section. The essential parameters can also be found in tabulated form in Table II. The information is plotted as the number of photons arriving in any bin normalized to the bin of most frequent arrival versus the time spread incurred. Note that the bins are logarithmic and therefore represent much less span in time to the left than to the right.

Figures 24, 25 and 26 compare time spread for NOSC Fog, Water Cloud C.2 and Henyey-Greenstein, respectively in a cloud one kilometer thick of optical thickness four. It can be seen by comparing the graphs that at this optical thickness the

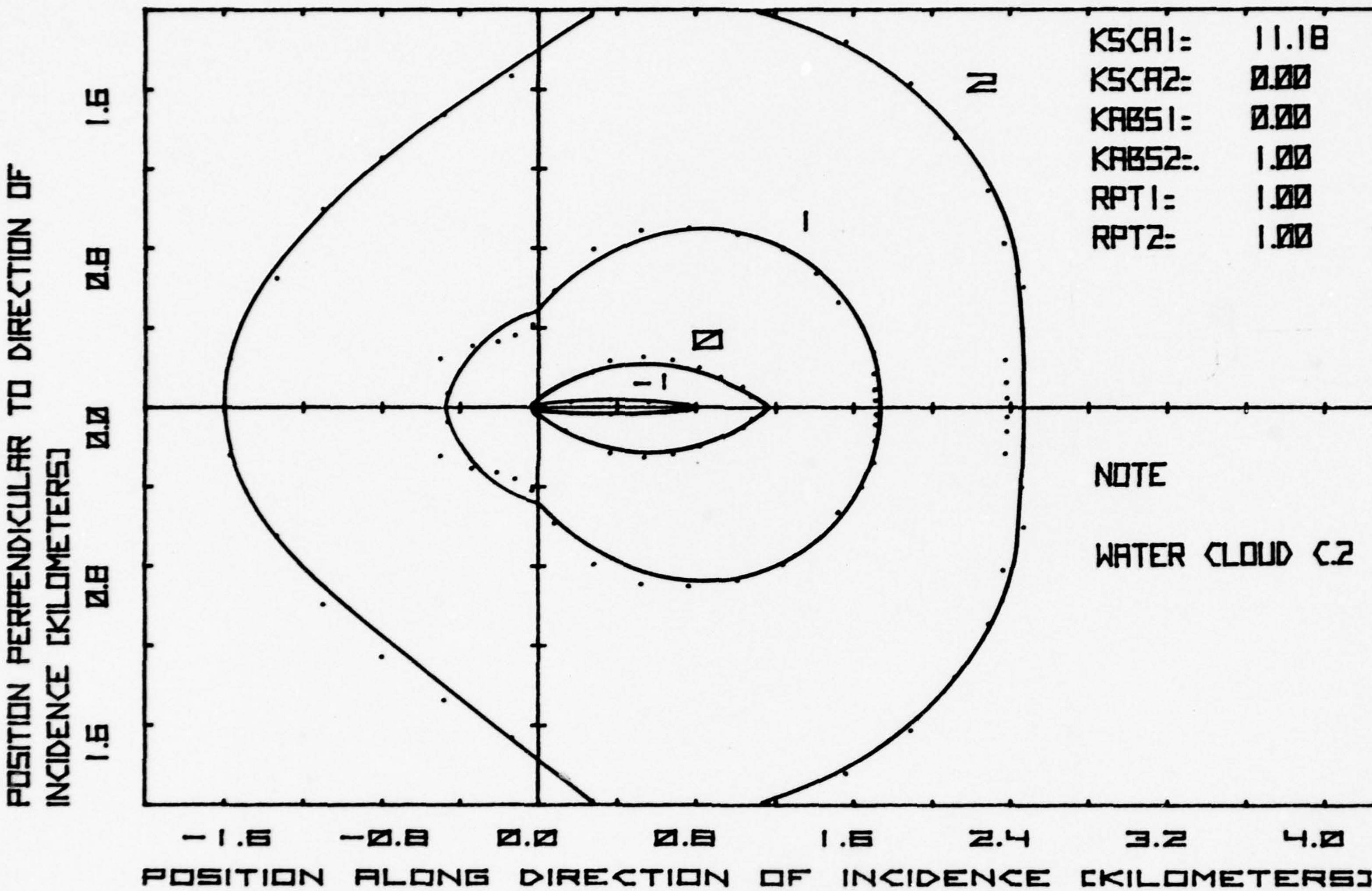


Figure 22

Spatial Spread Dependence on Phase Function

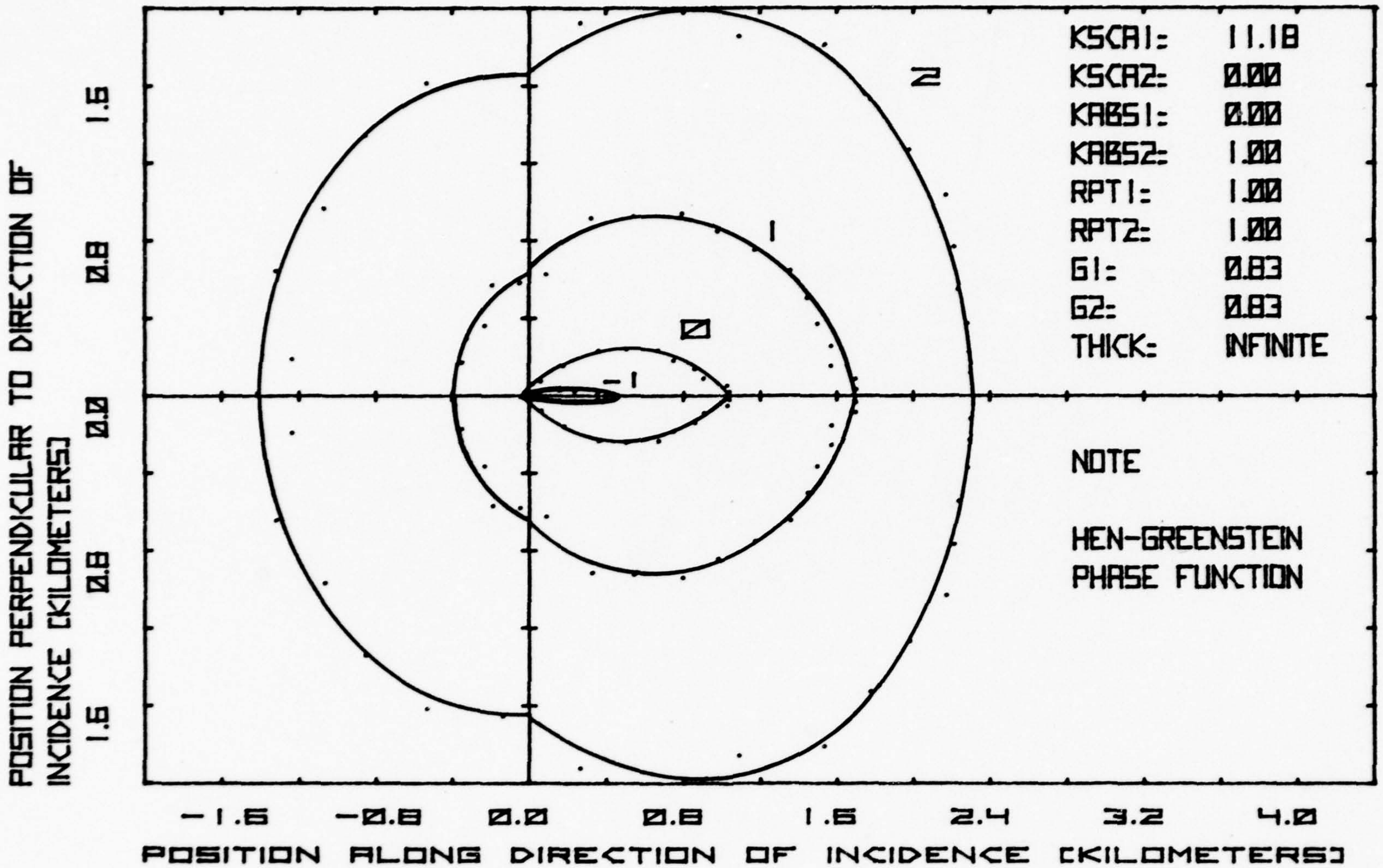


Figure 23

Spatial Spread Dependence on Phase Function

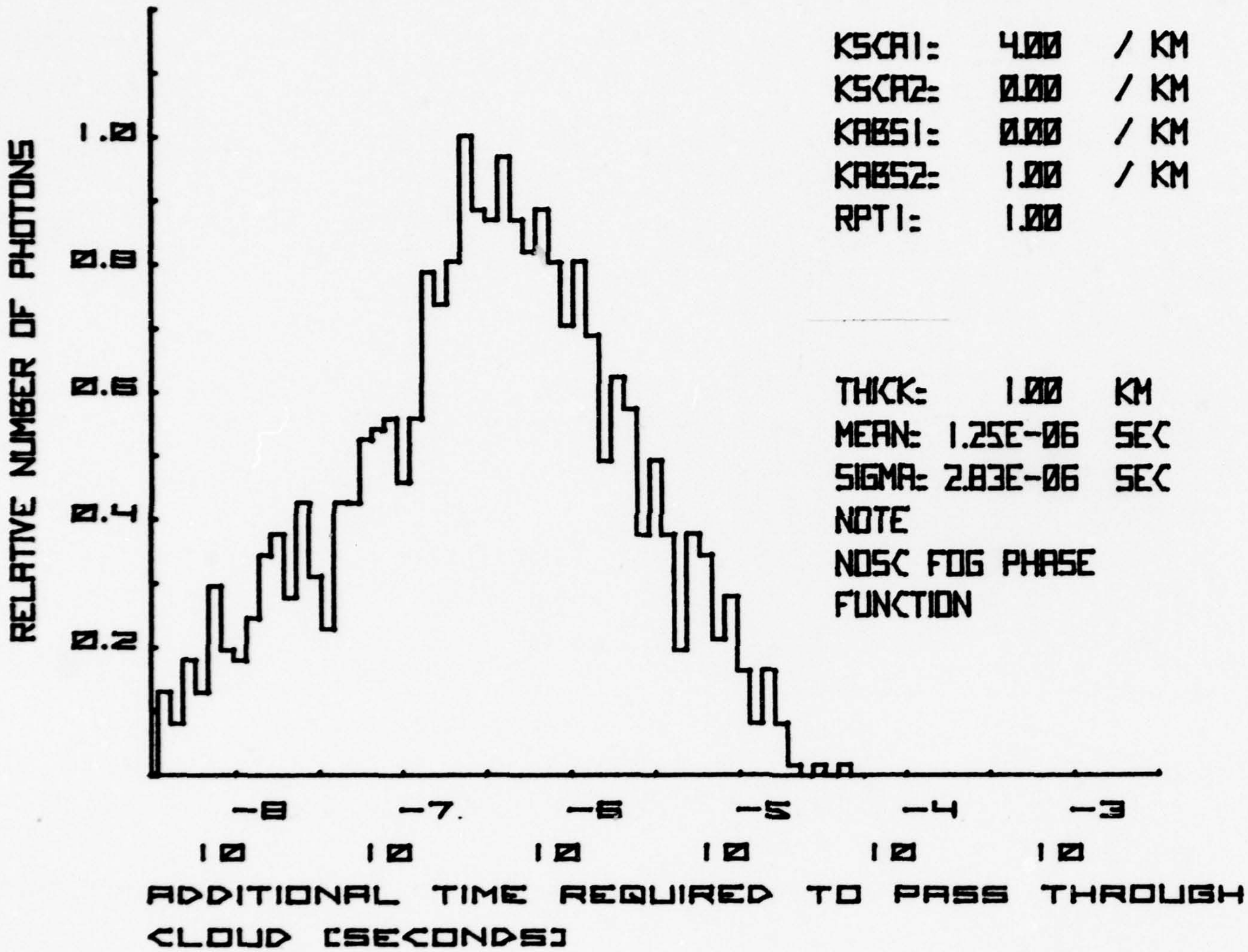


Figure 24

Time Spread Dependence on Phase Function

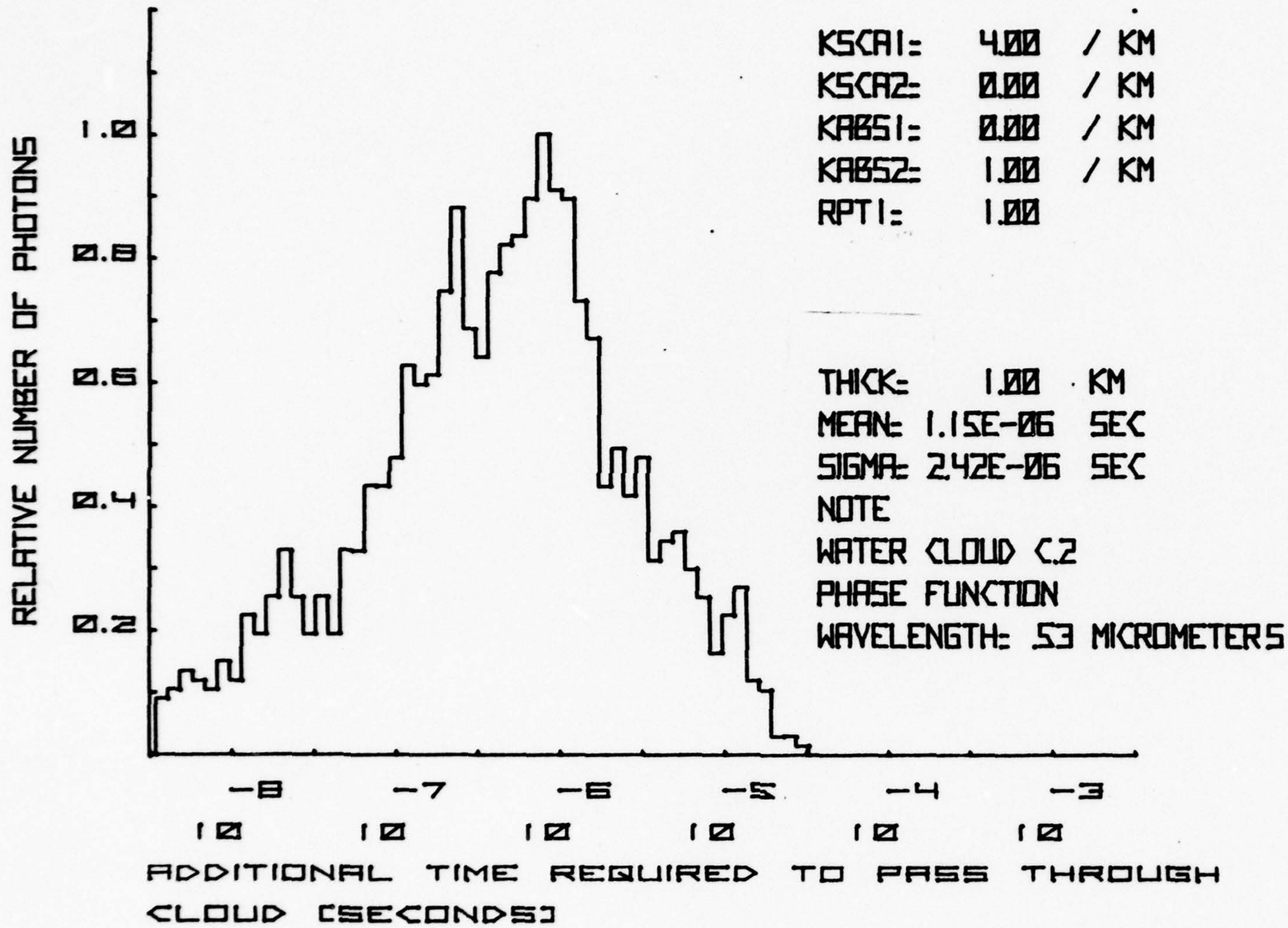


Figure 25

Time Spread Dependence on Phase Function

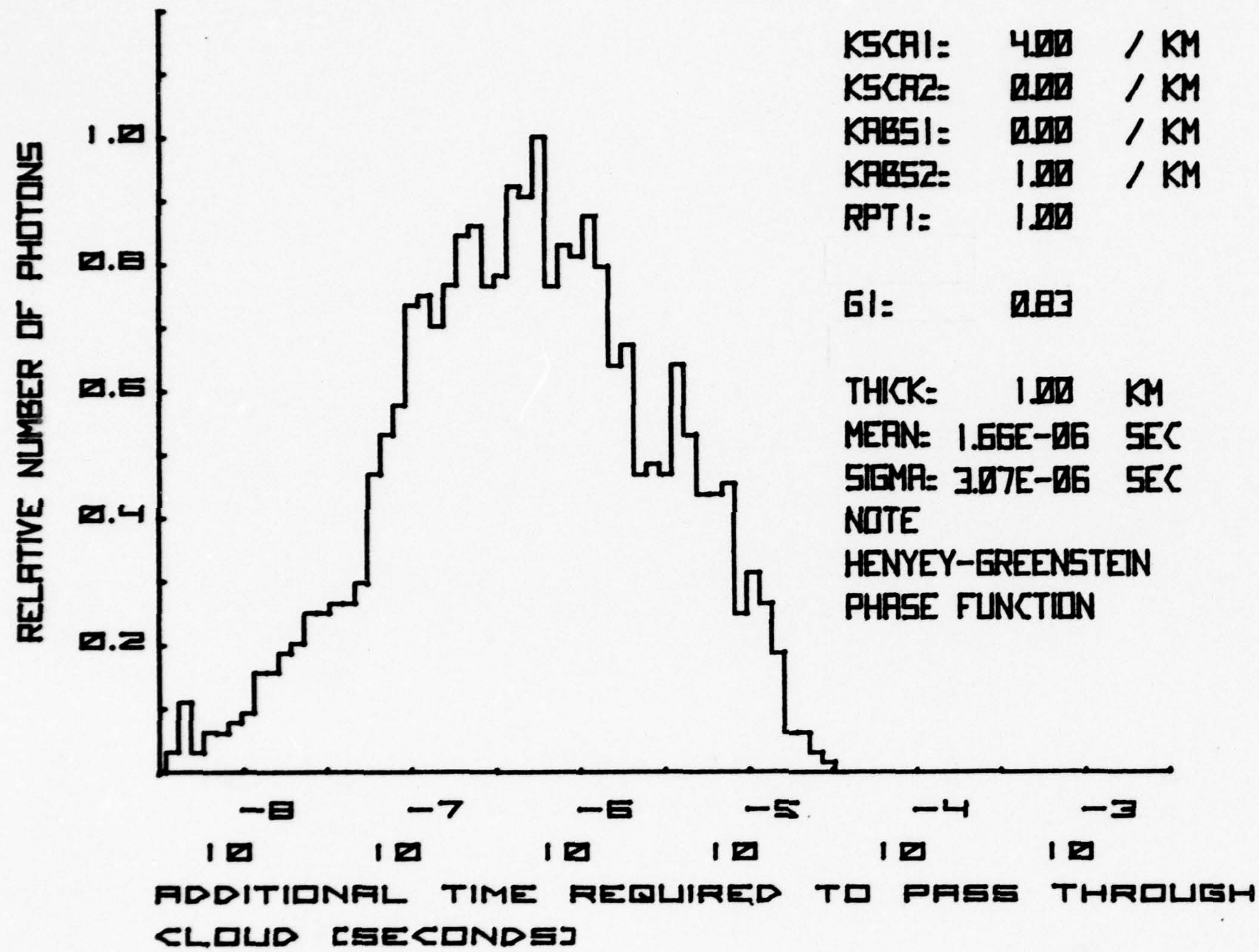


Figure 26

Time Spread Dependence on Phase Function

higher peaked phase functions have more photons arriving at earlier times.

Figures 27, 28 and 29 similarly compare time data for a cloud with a physical thickness of one kilometer and an optical thickness of eight. The time spread diagrams begin to look very much alike but still vary slightly with peakedness. At this thickness the average transit time is nearly identical for the three cases but there are still photons of the higher peaked phase functions which traverse the cloud with very little time delay. This indicates some dependence on phase function for this thickness.

Figures 30, 31 and 32 conclude the graphical time spread presentation with a cloud one kilometer thick with an optical thickness of 15. There is no noticeable difference in the graphs that can not be attributed to the statistical deviation expected in a Monte Carlo routine. Table III summarizes the time spread data.

From these results it can be said that the time spread becomes independent of the details of the phase function at optical thicknesses greater than 15.

C. REGION OF TRANSITION FROM FORWARD TO MULTIPLE SCATTER

The region of transition from forward to multiple scatter can be defined as that region for which (1) distances of lesser magnitude show time and spatial spread depending markedly on the phase function of the scattering particles and (2) distances of greater magnitude show that time and spatial spread do not depend on the phase function of the scattering particles.

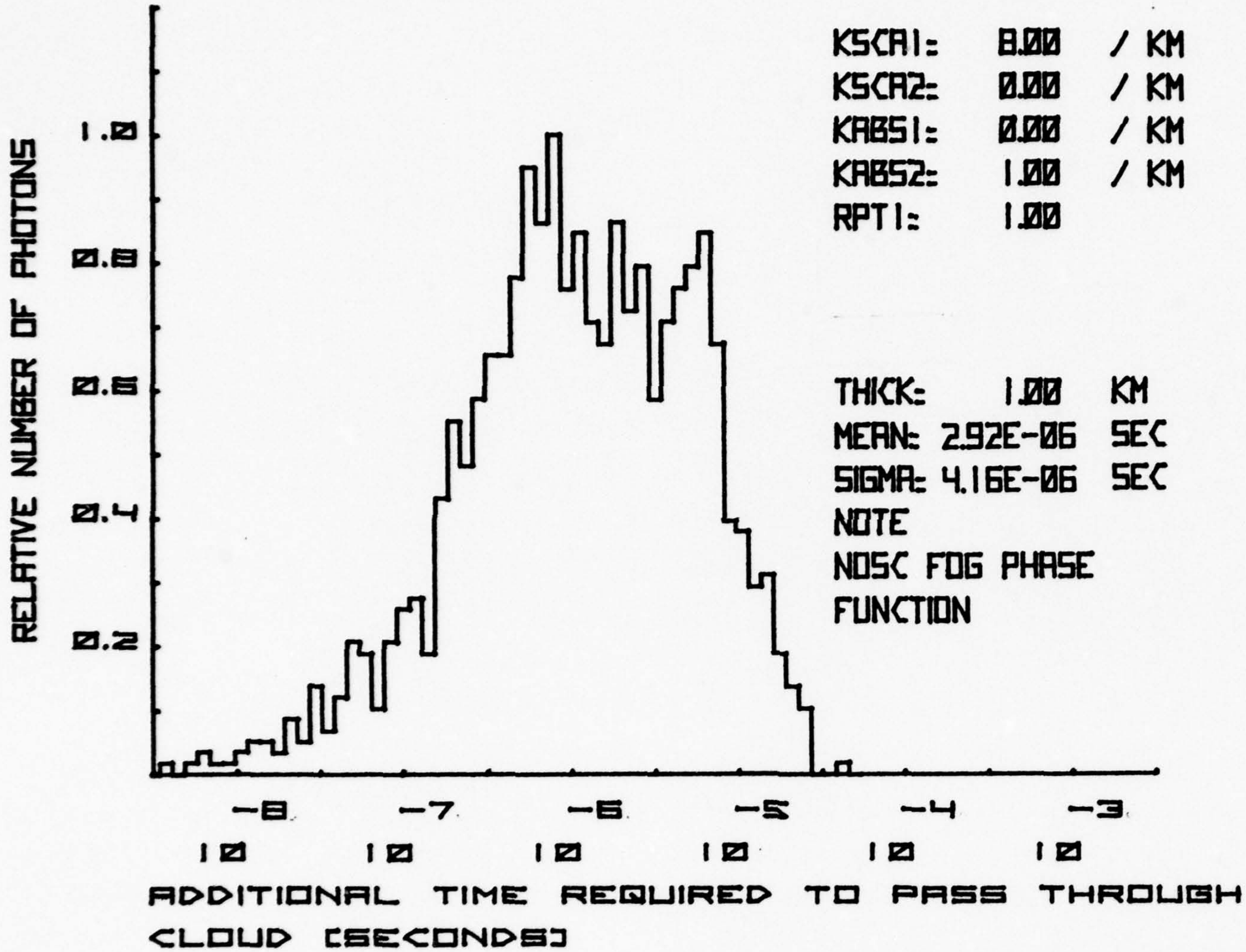


Figure 27

Time Spread Dependence on Phase Function

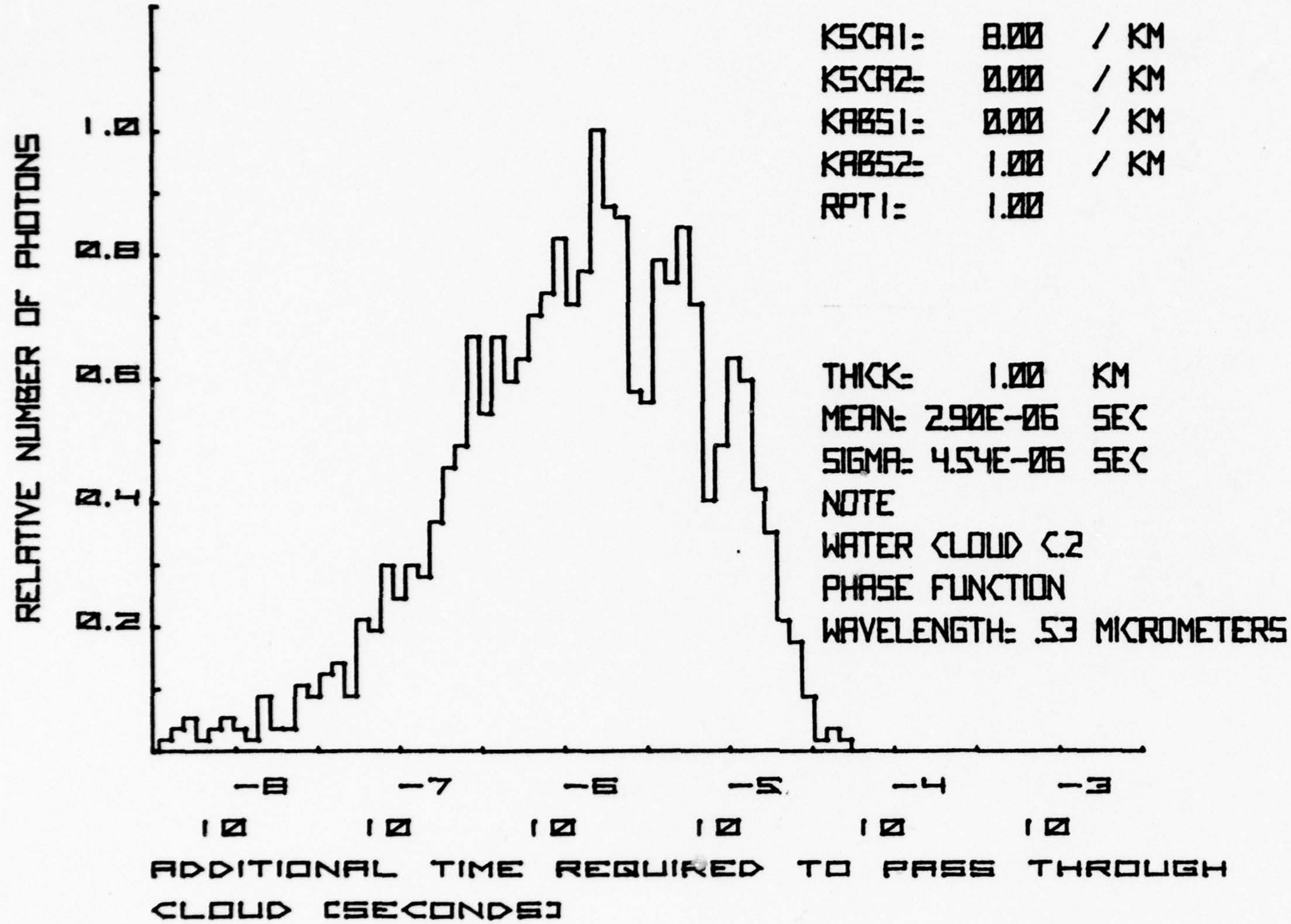


Figure 28

Time Spread Dependence on Phase Function

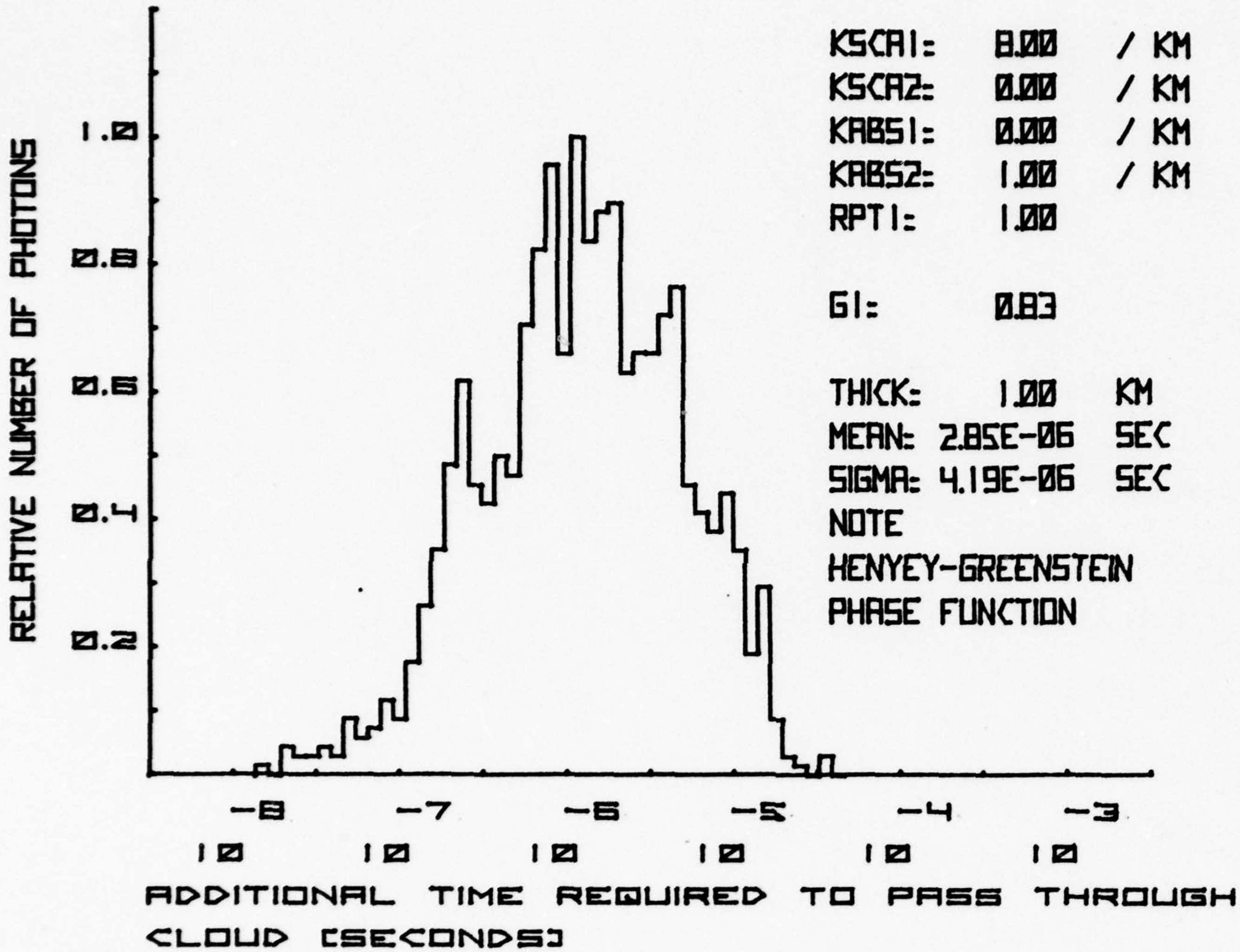


Figure 29

Time Spread Dependence on Phase Function

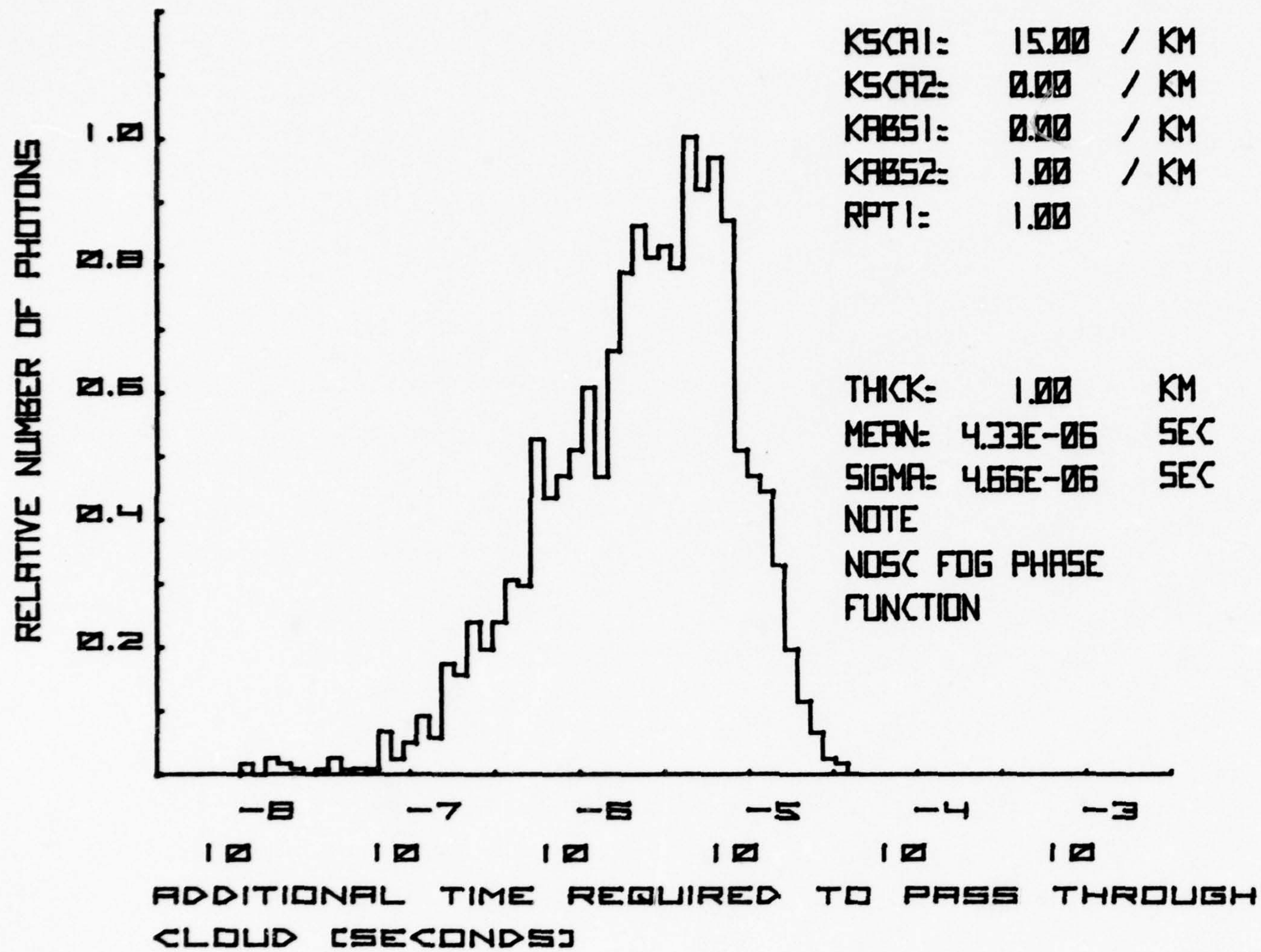


Figure 30

Time Spread Dependence on Phase Function

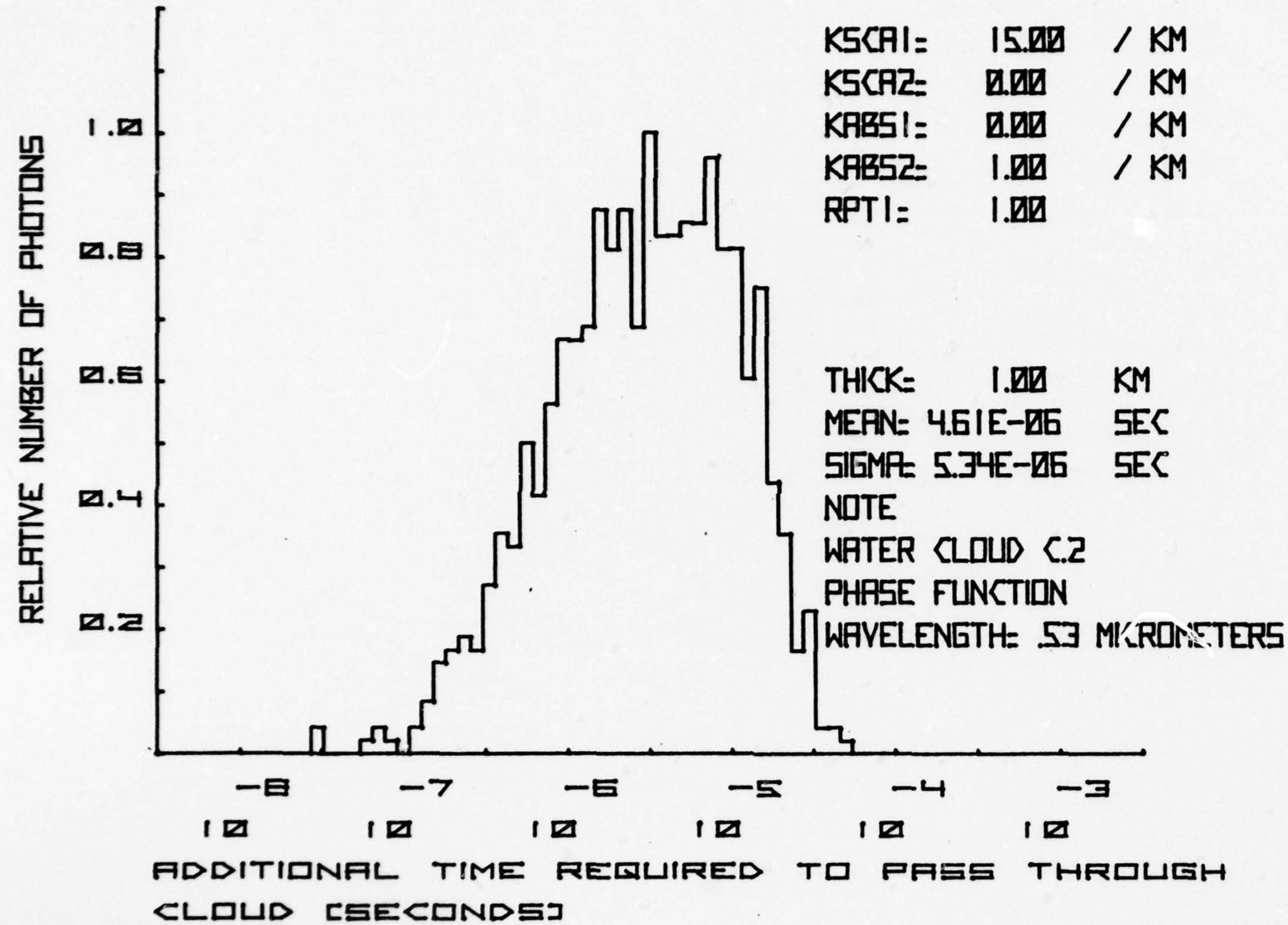


Figure 31

Time Spread Dependence on Phase Function

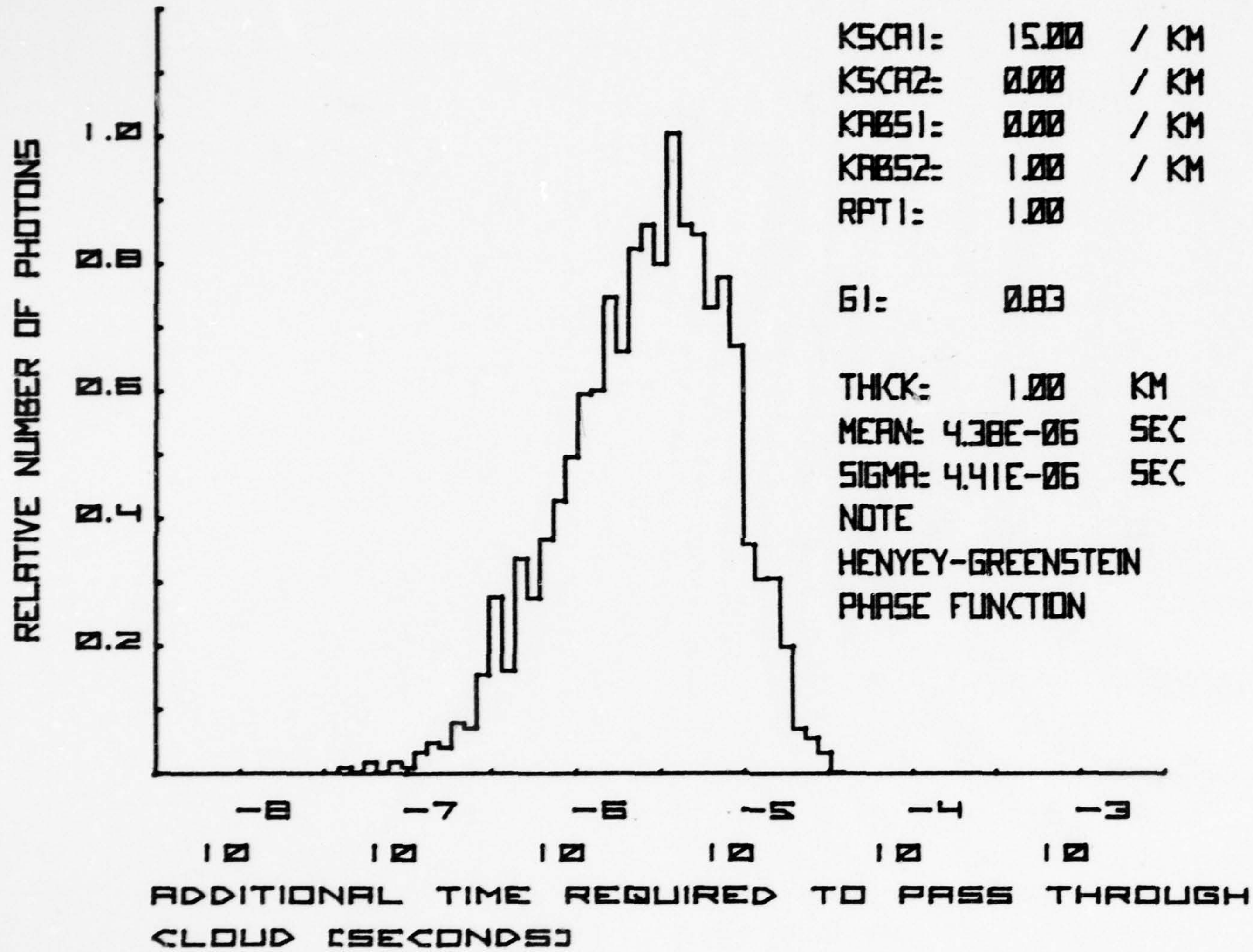


Figure 32

Time Spread Dependence on Phase Function

TABLE III

Summary of Time Spreading by One Kilometer Thick Cloud

KSCA1 (km^{-1})		H.G. G=.83	W.C. C.2 (μ-sec)	NOSC Fog
4	Mean	1.7	1.2	1.2
	Sigma	3.1	2.4	2.8
8	Mean	2.9	2.9	2.9
	Sigma	4.2	4.5	4.1
15	Mean	4.4	4.6	4.3
	Sigma	4.4	5.3	4.7
30	Mean	7.5	7.1	7.9
	Sigma	5.3	5.6	5.5

From the previous two sections the region of transition from forward to multiple scatter is between about 10 - 20 optical thicknesses. This agrees favorably with observations made by other authors [40].

D. EFFECT ON SPATIAL CHARACTER OF LIGHT WHEN PASSING THROUGH A CLOUD

In this section four graphs are used to display the effect of finite thickness clouds on the spatial spread of light. All four diagrams use the relative flux contours and the list of parameters defined in earlier sections of this report.

Figure 33 shows the relative flux contours of an infinite very clear atmosphere. Figure 34 shows the effect when a cloud of optical thickness about nine is placed in the light's path. Deformation of the contours is very obvious.

Likewise, Figures 35 and 36 show the effect of a .5 kilometer Water Cloud C.2 when placed in the path of light in a clear atmosphere. The sharp peaks on the flux contours are rounded and as expected, light is significantly redirected when passing through the cloud. Clouds of large optical thickness would eventually cause the Lambertian irradiance well known in the theory of light propagation.

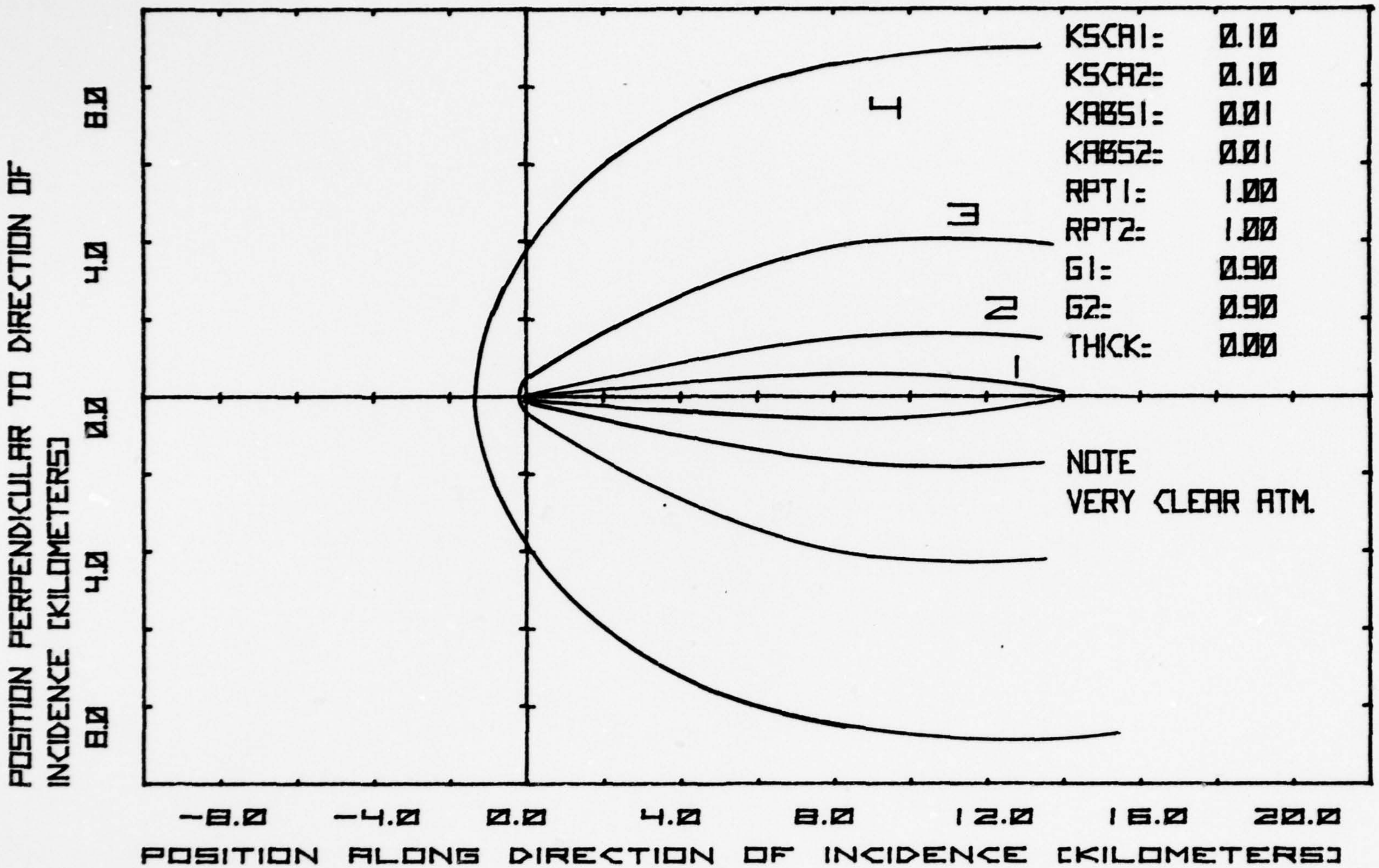


Figure 33

Effect of Finite Clouds on Spatial Character of Light

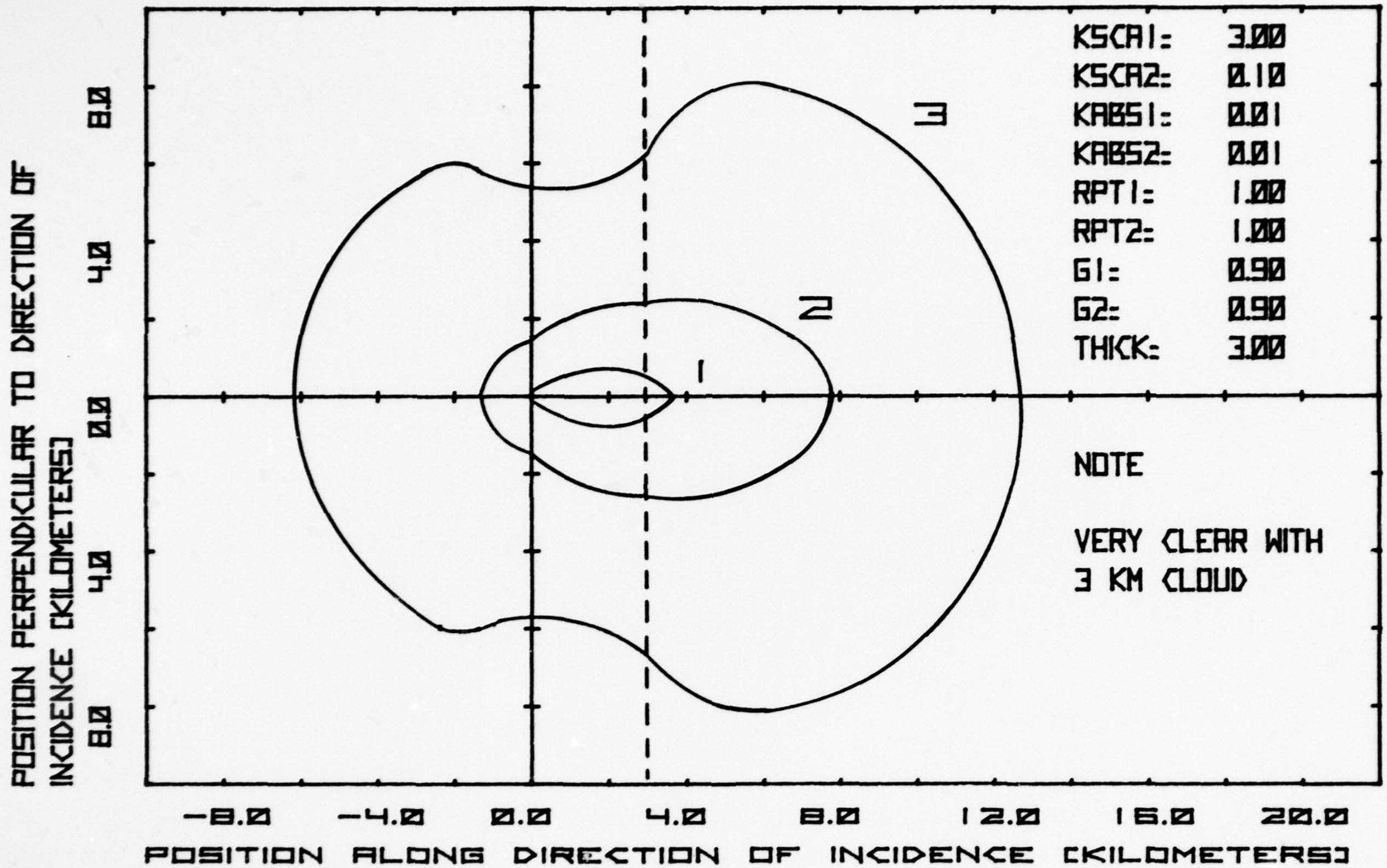


Figure 34

Effect of Finite Clouds on Spatial Character of Light

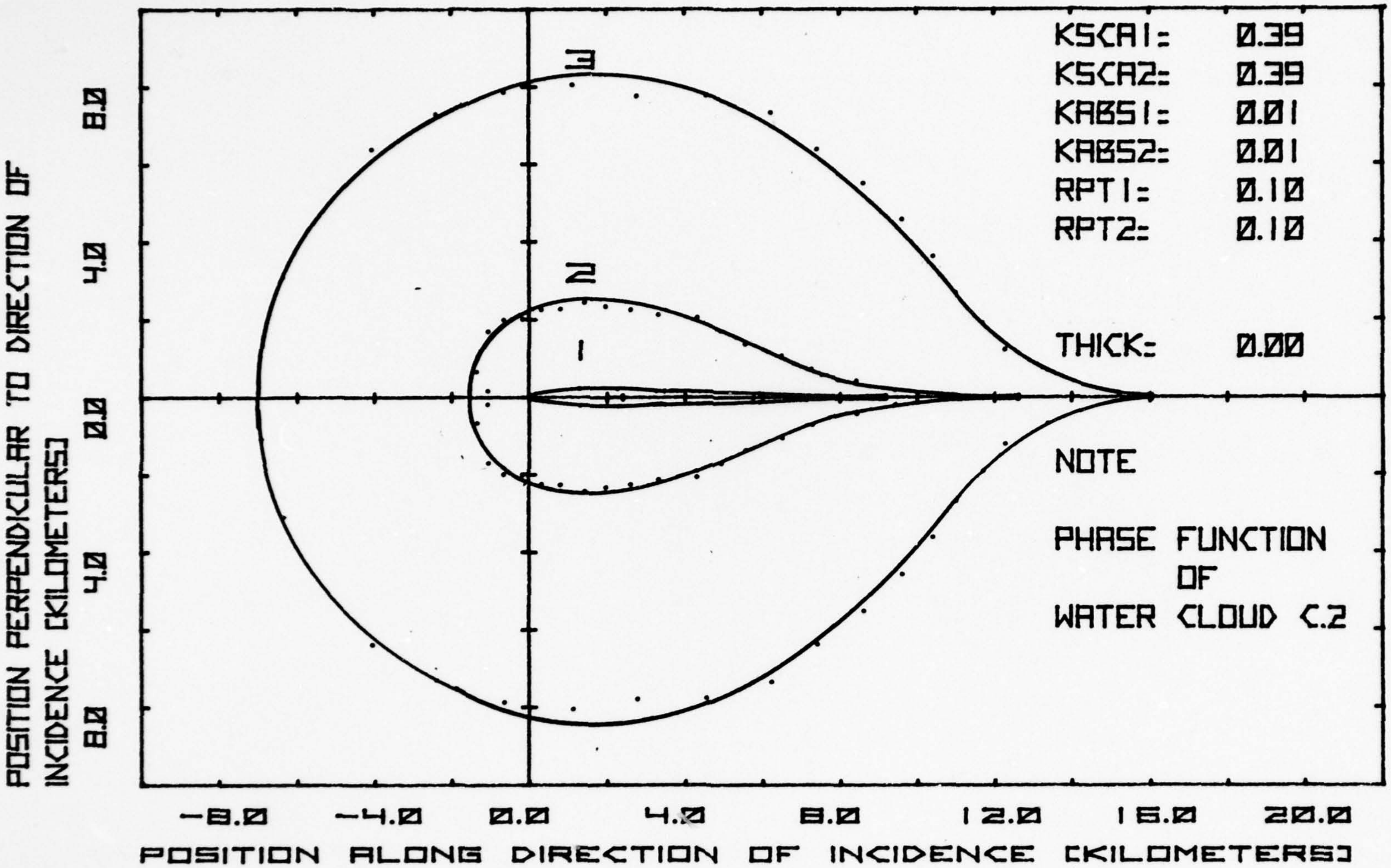


Figure 35

Effect of Finite Clouds on Spatial Character of Light

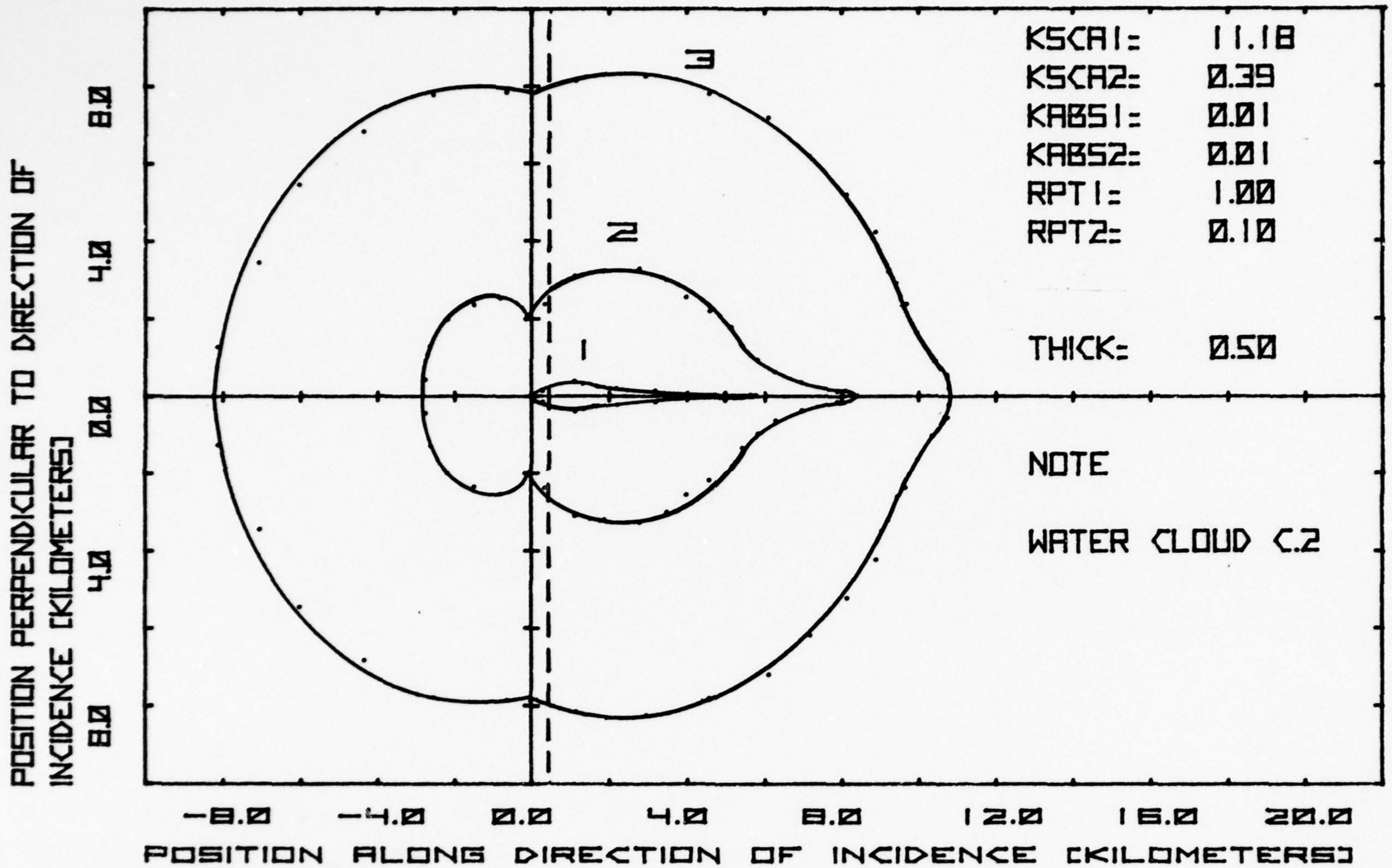


Figure 36

Effect of Finite Clouds on Spatial Character of Light

VI. DISCUSSION

In this thesis Monte Carlo methods and simple analytical methods were used to characterize light transmission through a multiple scattering medium. Complicated numerical methods were not used nor were highly mathematical models. This type of approach remains for future investigations. It is anticipated that results drawn from present methods will be confirmed.

Much information on the multiple scattering problem remains to be gathered. The present Monte Carlo routine can be adapted to numerous useful geometries and atmospheres with gradients and time variation of physical parameters. Propagation over and around solid bodies such as mountains and the earth itself can be modeled. Layered media such as cloud, air and water interfaces can also be simulated by the routine.

VII. CONCLUSIONS

A computer routine based on a Monte Carlo model was developed to simulate light propagation through a scattering absorbing medium. The routine can simulate a plane parallel cloud of scatterers with the desired parameters located within a medium with another set of parameters. The phase functions of each medium can be arbitrarily defined using a set of data pairs or can be approximated by closed form expressions. The data generated by the routine has been checked for accuracy against other theories using analytical methods. Each comparison shows adequate agreement between theories where agreement can be expected.

The routine will automatically plot spatial information necessary in characterizing light transfer through the medium by use of contours of equal photon flux. It will also generate histograms depicting time spread information for light passing through finite clouds.

Using the Monte Carlo routine created and inputting phase functions of different peakedness, it has been found that both spatial and temporal spread are independent of the details of the phase function for thicknesses greater than 15 extinction lengths. The region of transition from forward scatter to multiple scatter is between 10 - 20 extinction lengths.

The routine has also been used to study the effect finite thickness clouds have on the spatial character of light.

APPENDIX A

I. GENERATION OF A RANDOM SCATTER/ANGLE WEIGHTED BY AN ARBITRARY PHASE FUNCTION

A. STATEMENT OF THE PROBLEM

Reference 6 outlines the theory and calculations necessary to generate random numbers weighted in accordance with functions that represent characteristics of a scattering medium. At each collision new scatter angles and a distance were generated by closed form expressions which weight a uniformly distributed random number. However, in the more general case of a polydispersion, the computed phase function could not be represented adequately in closed form. This made it impossible to invert the function enabling analytical generation of a weighted scatter distribution. It was necessary to create a method for generation of a random theta weighted by an arbitrary phase function.

B. METHOD USED IN WEIGHTING THETA

Reference 21 and Appendix B explain in great detail a method for calculating the representative phase function of a polydispersion given the particle size distribution and composition. Using this computer adaptation of Mie theory, values of the averaged normalized phase function were generated at selected angles of scatter. Given this data and a random number, R, weighted uniformly over the closed interval [0, 1], the problem is to solve the following equation for θ_R.

$$R = \frac{\int_0^{\theta_R} P(\theta)\sin\theta\, d\theta}{\int_0^{\pi} P(\theta)\sin\theta\, d\theta} \qquad (1)$$

where $P(\theta)$ is the averaged normalized phase function and θ is the variable of integration. Notice that,

$$\theta = 0 \text{ when } R = 0, \qquad (2)$$
$$\theta = \pi \text{ when } R = 1,$$

as you would expect on the closed interval. $P(\theta)$ is not a continuously defined function over the interval $[0, \pi]$ in this case so the integrals are represented numerically by the trapezoidal rule. Figure 37 diagrams the basic procedure used. N values of $P(\theta)$ are selected so as to closely approximate the phase function. Of course, more values are selected near the small scatter angles to adequately describe the sharp peak. The N values are multiplied by the sine of the respective scatter angle ($P(\theta)$ includes $\sin\theta$ implicitly in Figure 37). N-1 trapezoids are established using these N values. The area of each trapezoid is divided by the total area of all trapezoids thus establishing a weight for the respective interval.

$$W_i = \frac{(P_i\sin\theta_i + P_{i+1}\sin\theta_{i+1})(\theta_{i+1} - \theta_i)}{\sum_{i=1}^{N-1}(P_i\sin\theta_i + P_{i+1}\sin\theta_{i+1})(\theta_{i+1} - \theta_i)} \qquad (3)$$

This of course requires that

$$\sum_{n=1}^{N-1} W_n = 1. \qquad (4)$$

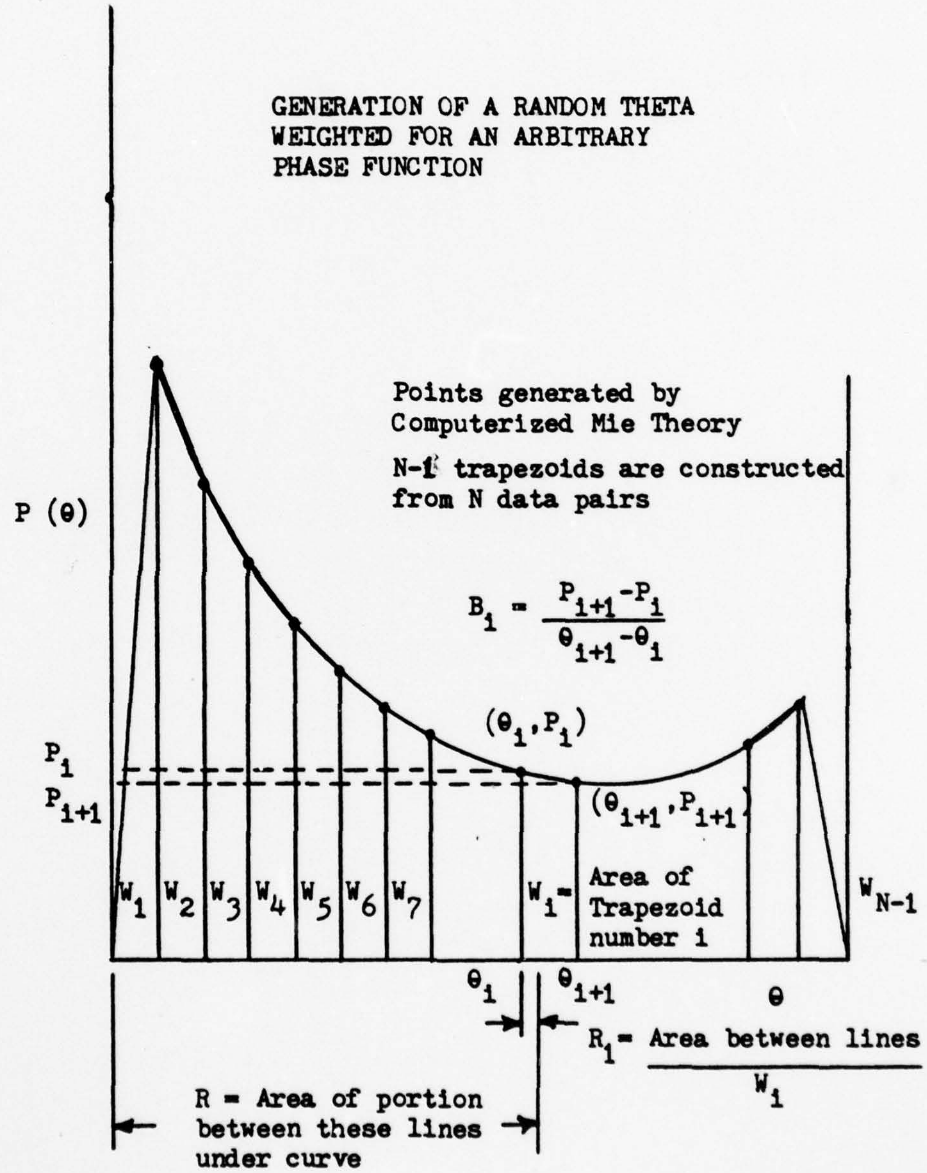


Figure 37

Diagram of Random Generation of Theta

With the weight of each panel known, the uniform random number R is used to solve for the first value of n such that,

$$\sum_{i=1}^{n} W_i > R \qquad (5)$$

is satisfied. The random θ_R required is somewhere between θ_n and θ_{n+1}. Within the panel, θ is weighted linearly in a fashion as explained in Ref. 6. For a given panel, θ_i, θ_{i+1}, P_i and P_{i+1} are known. From these the slope and intercept are estab-lsihed.

$$B_i = \frac{P_{i+1}-P_i}{\theta_{i+1}-\theta_i} \qquad A_i = P_i - B_i\theta_i \qquad (6)$$

The panel is normalized as follows:

$$\mathrm{NORM}\int_{\theta_i}^{\theta_{i+1}} (A_i+B_i\theta)d\theta = 1 \qquad (7)$$

so that,

$$\mathrm{NORM} = \frac{1}{A_i(\theta_{i+1}-\theta_i)+\frac{B_i}{2}(\theta_{i+1}{}^2-\theta_i{}^2)} \qquad (8)$$

All that remains is to solve the following for θ_R,

$$R_1 = \frac{(R-\sum_{i=1}^{n} W_i)}{(\sum_{i=1}^{n+1} W_i - \sum_{i=1}^{n} W_i)} = \mathrm{NORM}\int_{\theta_i}^{\theta_R} (A_i+B_i\theta)d\theta \qquad (9)$$

After a few algebraic steps, the solution is

$$\theta_R = \frac{-A_i}{B_i} \pm \left(\left(\frac{A_i}{B_i}\right)^2 + \left(\frac{2C_i}{B_i}\right) \right)^{\frac{1}{2}} \quad \begin{cases} + & B_i > 0 \\ - & B_i < 0 \end{cases} \tag{10}$$

where,

$$C_i = R_1\left(A_i\theta_{i+1} + \frac{B_i\theta_i^{\,2}}{2}\right) + [1 - R_1]\left(\frac{B_i\theta_i^{\,2}}{2} + A_i\theta_i\right) \tag{11}$$

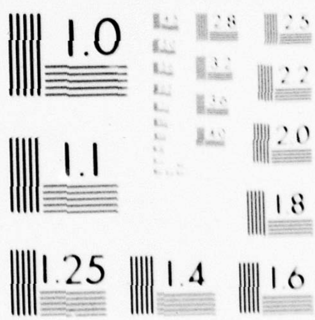
This θ_R has the desired properties. The FORTRAN coding of this method is found in the Subroutine RANTH.

AD-A057 991 NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF F/G 20/6
COMPUTER SIMULATION OF LIGHT PROPAGATION THROUGH A SCATTERING M--ETC(U)
JUN 78 M A MILLBACH

UNCLASSIFIED NL

2 OF 2
AD A057 991

END
DATE FILMED
10-78
DDC



MICROCOPY RESOLUTION TEST CHART

APPENDIX B

I. DESCRIPTION AND DOCUMENTATION OF PROGRAM TO ADAPT MIE THEORY TO MACHINE COMPUTATION

A. INTRODUCTION

Mie theory has been adapted to machine computation on many occasions. Deirmendjian [21] provides an excellent guideline for using Mie theory on spherical polydispersions. Using Deirmendjian's guideline, a computer routine was created to generate the volume scattering and extinction cross sections and the corresponding elements of the normalized scattering matrix for a polydispersion where the number of particles per unit volume, per unit radius is given by,

$$n(r) = ar^{\alpha} e^{-br^{\gamma}} \qquad o<r<\infty \qquad (1)$$

where r is the particle radius. The four constants a, α, b and γ are positive and real and α is an integer. They are not independent of each other, and are related to quantities in the particle frequency distribution. The radius which is most frequently found in the particle distribution is r_c and N is the total number of particles per unit volume. Both N and r_c can be found by experimental measurement. In terms of N and r_c the constants of the distribution are found using:

$$N = a\int_0^{\infty} r^{\alpha} e^{-br^{\gamma}} dr = \frac{a}{\gamma b^{\frac{(\alpha+1)}{\gamma}}} \Gamma\left(\frac{\alpha+1}{\gamma}\right) \qquad (2)$$

$$b = \frac{\alpha}{\gamma r_c^{\gamma}} \quad (3)$$

and by choice of α and γ to best fit the experimental distribution function. Γ is the usual gamma function.

B. EQUATIONS TO BE CALCULATED

The equations used to generate scattering data for a polydispersive system are extensions of those used in a monodispersion. The distribution has been created in a manner requiring that

$$N = \int_{r_1}^{r_2} n(r)dr \quad (4)$$

where n(r) is a continuous and integrable function within the range and represents the partial concentration per unit volume per unit increment of radius r. The values of interest are volume scattering cross sections and the corresponding elements of the normalized scattering matrix [18, 21]. These values can be computed using

$$\beta_{sca}[\lambda, n(x)] = \pi k^{-3}\int_0^{\infty} x^2 n(x)Q_{sca}(x)dx \quad (5)$$

and

$$\beta_{ext}[\lambda, n(x)] = \pi k^{-3}\int_0^{\infty} x^2 n(x)Q_{ext}(x)dx \quad (6)$$

$$P_j(\theta) = \frac{4\pi}{k^3\beta sca}\int_0^{\infty} n(x)\ i_j(\theta)dx \quad j=1,2 \quad (7)$$

where

$Q_{sca}(x)$ = scattering efficiency factor

$Q_{ext}(x)$ = extinction efficiency factor

$i_j(x)$ = dimensionless intensity parameters

Each of the above terms will be examined in detail in following sections.

C. CALCULATION OF SCATTERING EFFICIENCY FACTORS AND DIMENSIONLESS INTENSITY PARAMETERS

To perform numerical integration, the integrand must be evaluated at many different values within the range of integration. In this case the scattering and extinction efficiency factor must be evaluated for numerous Mie size parameters x. Scattering efficiency is a name given to the ratio of total scattering cross section to geometrical cross section. A similar definition is used to define extinction efficiency. For any particle size and composition, Mie theory gives the total scattering cross section as [18, 21],

$$\delta_{sca}(m,x) = \frac{1}{2}\int_{\Omega} (A_1A_1^{*}+A_2A_2^{*})d\omega \tag{8}$$

where

$$kA_1 = S_1(m,x,\theta) = \sum_{n=1}^{\infty} \frac{2n+1}{n(n+1)}(a_n\pi_n+b_n\tau_n) \tag{9}$$

$$kA_2 = S_2(m,x,\theta) = \sum_{n=1}^{\infty} \frac{2n+1}{n(n+1)}(b_n\pi_n+a_n\tau_n) \tag{10}$$

using the complex index of refraction of the particle m, the Mie coefficients a_n, b_n and angular coefficients π_n, τ_n, each of which is described in detail in later sub-sections. Using these equations, the scattering efficiency factors are [18, 21],

$$Q_{sca}(m,x) = \frac{2}{x^2}\sum_{n=1}^{\infty}(2n+1)(|a_n|^2+|b_n|^2), \tag{11}$$

$$Q_{ext}(m,x) = \frac{2}{x^2}\sum_{n=1}^{\infty}(2n+1)\mathrm{Re}\{a_n+b_n\}. \tag{12}$$

The dimensionless intensity parameters are given by the expressions,

$$\begin{aligned} i_1(x,m,\theta) &= k^2 A_1 A_1^* = S_1 S_1^* \\ i_2(x,m,\theta) &= k^2 A_2 A_2^* = S_2 S_2^* \end{aligned} \tag{13}$$

where A_1, A_2, S_1 and S_2 are the same as above. The procedure used in calculating a_n, b_n, τ_n and π_n is described next.

1. Computation of τ_n and π_n

The two angular functions τ_n and π_n are required before the dimensionless intensity parameters can be calculated. τ_n and π_n are functions of $\mu = \cos\theta$ only and as the subscript implies, there are actually many different angular functions. Both functions are defined in terms of Legendre polynomials [18] and their derivatives. The following recursion relations are used in calculating τ_n and π_n,

$$\pi_n(\theta) = \cos\theta \frac{2n-1}{n-1} \pi_{n-1}(\theta) - \frac{1}{n-1} \pi_{n-2}(\theta) \qquad (14)$$

$$\tau_n(\theta) = \left[\cos\theta \left[\pi_n(\theta) - \pi_{n-2}(\theta)\right]\right] - \left[(2n-1)\sin^2\theta \; \pi_{n-1}(\theta)\right] + \tau_{n-2}(\theta) \qquad (15)$$

with

$$\begin{array}{ll} \pi_o(\theta) = 0 & \tau_o(\theta) = 0 \\ \pi_1(\theta) = 1 & \tau_o(\theta) = \cos\theta \qquad (16) \\ \pi_2(\theta) = 3\cos\theta & \tau_2(\theta) = 3\cos 2\theta \end{array}$$

2. Computation of a_n and b_n

The Mie coefficients a_n and b_n have a complex dependence on the index of refraction of the particle, and the surrounding medium and also depend on the size parameter, x. There exists many different expressions for the Mie coefficients in terms of known mathematical functions. The form used in this program is described in terms of spherical Bessel functions each of which is described in later sections. The relations for a_n and b_n are

$$a_n = \frac{\mu\left[yj_{n-1}(y) - nj_n(y)\right] xj_n(x) - \varepsilon yj_n(y)\left[xj_{n-1}(x) - nj_n(x)\right]}{\mu\left[yj_{n-1}(y) - nj_n(y)\right] xh_n^2(x) - \varepsilon yj_n(y)\left[h_{n-1}^2(x) - nh_n^2(x)\right]} \qquad (17)$$

$$b_n = \frac{\varepsilon\left[yj_{n-1}(y) - nj_n(y)\right] xj_n(x) - \mu yj_n(y)\left[xj_{n-1}(x) - nj_n(x)\right]}{\varepsilon\left[yj_{n-1}(y) - nj_n(y)\right] xh_n^2(x) - \mu yj_n(y)\left[h_{n-1}^2(x) - nh_n^2(x)\right]} \qquad (18)$$

where μ is the absolute value of the complex index of refraction outside the sphere, ε is the absolute value of the complex index of refraction inside the sphere, $x = 2\pi r/\lambda$ and $y = mx$. The functions j_n, y_n, $h_n^2 = j_n - iy_n$ are the spherical Bessel and Neuman functions. A procedure for calculating their values is outlined in the following sections.

3. Computation of j_n, y_n, h_n^2

The argument of these Bessel related functions is complex in some cases, so much care is required in their calculation. Each function is calculated using recursion relations and the lowest order functions as follows. For j_n,

$$j_{n+1}(z) = \frac{(2n+1)j_n(z)}{Z} - j_{n-1}(z) \tag{19}$$

$$j_o(z) = \frac{\sin z}{z}$$

$$j_1(z) = \frac{\sin z}{z} - \frac{\cos z}{z} \tag{20}$$

$$j_2(z) = \left(\frac{3}{z^3} - \frac{1}{z}\right)\sin z - \frac{3}{z^2}\cos z$$

likewise for y_n,

$$y_{n+1}(z) = \frac{(2n+1)y_n(z)}{z} - y_{n-1}(z) \tag{21}$$

$$y_o(z) = \frac{-\cos z}{z}$$

$$y_1(z) = \frac{-\cos z}{z} - \frac{\sin z}{z} \tag{22}$$

$$y_2(z) = \left(\frac{-3}{z^3} + \frac{1}{z}\right)\cos z - \frac{3}{z^2}\sin z$$

and these two yeild h_n^2,

$$h_n^2(z) = j_n(z) - iy_n(z) \tag{23}$$

Because the argument of the sine and cosine in evaluating the spherical Bessel function is complex, the following relations are needed,

$$\begin{aligned} z &= a + ib \\ \cos z &= \cos(a)\cosh(b) - i\sin(a)\sinh(b) \qquad (24) \\ \sin z &= \sin(a)\cosh(b) + i\cos(a)\sinh(b) \end{aligned}$$

where cosh and sinh are the usual hyperbolic trigonometric functions.

D. DESCRIPTION AND DOCUMENTATION OF PROGRAM

The program adapting Mie theory to machine computation is composed of the MAIN routine and numerous subroutines described in the following sub-sections.

1. MAIN Routine

The MAIN routine handles the input and output functions necessary in program operation. The input parameters include indices of refraction inside and outside of the spheres, constants of the particle distribution and the most frequently occurring value of Mie size parameter in the distribution, x_c. Through x_c, the wavelength of the incident light is entered because r_c is known for any desired particle distribution.

Other necessary inputs are the number of scatter angles desired in the output scattering matrix elements $P_j(\theta)$, the smallest value of x, the increment in x to be used for numerical integration and the odd number of x values at which the integrand is to be evaluated. The MAIN program accepts the input values and uses the distribution parameters to assign each designated x value a weight. Because the particle distribution is normalized, the sum of all weights assigned is unity.

Simpson's 1/3 rule is used to evaluate integrals 5, 6 and 7. At each value of x the scattering and extinction efficiency factors of equations 11 and 12 are calculated through use of a subroutine MIEM. Similarly, at each desired scattering angle the dimensionless intensity parameters of 13 are calculated through MIEM. Printout can be called at each value of x if so desired. The MAIN program cumulatively sums the areas of an even number of panels to get the desired results which are then printed.

2. MIEM Subroutine

MIEM is a subroutine composed to calculate the scattering and extinction efficiency factors in the integrand of equations 5 and 6 and the dimensionless intensity parameters of equation 13. Required inputs of MIEM, transferred to it by MAIN, are the index of refraction of the particle (henceforth the index of refraction of the outside medium will be unity), the size parameter x, and the number of scattering angles at which calculation of the intensity parameters is desired. MIEM

uses equations 11 and 12 to calculate the efficiency factors and equation 13 to calculate the intensity parameters. Because equations 9, 10, 11 and 12 require summation of series, each element of the series must be evaluated in turn. The terms involved in finding the efficiency factors are a_n and b_n which are calculated for each n = 1, 2, 3, 4,..... until the next term of the series is adequately small or the total number of terms exceeds 120. After each calculation of a_n and b_n its contribution to $Q_{sca}(x)$ and $Q_{ext}(x)$ is added to the previous total. MIEM uses function subroutines JN, FN and HN to calculate each of the a_n's and b_n's. Arrays JX, FX and HX are used to store values of the corresponding spherical Bessel functions during each recursion step.

After all the parameters a_n and b_n are calculated, MIEM turns them over to ANGLE to complete the remaining dimensionless intensity parameter calculations.

3. ANGLE Subroutine

ANGLE is a subroutine called by MIEM to calculate the dimensionless intensity parameters at each desired angle of scatter. ANGLE requires as inputs the Mie scattering coefficients generated in MIEM and the number of scatter angles desired. When called, ANGLE uses equations 9 and 10 to find S_1 and S_2 at each scatter angle. To do so ANGLE evaluates τ_n and π_n using recursion relations 14 and 15 with initial order functions, 16. At each n, the corresponding Mie coefficients a_n and b_n are used with τ_n and π_n and their contributions to S_1 and S_2 are added.

The process is repeated for each scatter angle and the resulting arrays are used to evaluate equation 13 at each scatter angle. The resulting dimensionless intensity parameters are then returned to MIEM and in turn to MAIN for integration using equation 7.

4. JN, FN, HN, CCOS, CSIN Complex Functions

These functions are called by MIEM in evaluation of the Mie coefficients a_n and b_n. JN, FN and HN contain logic to perform the recursion operation of equations 19 through 23. CCOS and CSIN are complex trigonometric functions drawn upon as needed by JN, FN and HN.

E. CPU TIME CONSIDERATIONS

CPU time depends largely on the wavelength to particle size ratio. This, of course, varies with the particle size distribution used. Thus, if the distribution includes many particles of large size compared to the wavelength the CPU time is great and vice versa. Typical CPU time requirements were on the order of 15 to 30 minutes for wavelengths of .53 to .28 microns, with about 300 x values in a distribution of Water Cloud C.2 of Deirmendjian [21]. The dependence on particle size to wavelength ratio is due to the fact that many terms are required for convergence of series for large values of x. As expected, CPU time goes up quite linearly with the number of x values used as integration points. There is little dependence on the number of scatter angles required.

APPENDIX C

I. SIMULATION OF A CLOUD IN MONTE CARLO ROUTINE LITE

A. INTRODUCTION

Reference 6 describes Monte Carlo simulation of light propagation through an infinite, homogeneous atmosphere. Many problems can be sufficently investigated using this model, but one of the advantages of a Monte Carlo model is its relative ease in adaptation to irregular geometries and inhomogeneous atmospheres. This Appendix gives one method for simulation of a cloud with plane parallel homogeneous medium. All macroscopic properties are the same everywhere inside of the cloud and another set of macroscopic parameters are the same everywhere outside of the cloud. This model circumvents the entire problem of describing and locating boundaries and inhomogeneities in real clouds. Figure 5 depicts the general structure of the cloud model giving the names of various parameters of the computer simulation. A point source of light is incident normal to the left edge of the cloud and its path is randomly generated until it exits the outermost sphere of the reference volume. Reference 6 describes the random path generation and accountability also used in this model so the terminology of that reference is used here for continuity whenever possible.

B. MODELING CLOUD BOUNDARIES AND PROPERTIES

The sets of parameters needed in defining two different scattering media are implemented by use of storage arrays. A binary logic code is used that switches whenever a photon crosses a boundary. Each array of characteristic parameters has two columns, one for inside the cloud and one for outside the cloud. The logic switch determines at each collision from which column the parameter is to be drawn.

Upon crossing a boundary, the photon is stopped and a new distance is randomly generated in accordance with the parameters of the newly entered medium. The method used for determination of whether or not a boundary was crossed is described in the following section.

C. DETERMINATION OF BOUNDARY CROSSING

The following equation is used to determine at each collision the angle between the original direction of incidence and the present position vector, $\bar{R}$,

$$\theta = \cos^{-1} \frac{\bar{R}\cdot(\bar{R}-\bar{R}_k)}{r}$$

where r is the distance between the collision point and the point of incidence on the cloud. $\bar{R}_k$ is an orientation vector from the point of collision to the (0, 0, 1) point of the fixed coordinate system. The r and θ values are computed at each collision which allows, due to cylindrical symmetry, calculation of the projected distance along the direction of incidence.

This distance is

$$\text{Projected distance} = r \cos \theta = DIST_{new}$$

which is compared to the thickness of the cloud. The medium in which the photon exists is determined by,

$$\left.\begin{array}{rll} & DIST_{new} < 0 & \\ 0 < & DIST_{new} < \text{THICKNESS} & \\ & DIST_{new} > \text{THICKNESS} & \end{array}\right\} \begin{array}{l} \text{Behind} \\ \text{Inside} \\ \text{Beyond} \end{array}$$

The location of the collision relative to the cloud is compared to the location of the previous collision and a boundary crossing is found if it has occurred between collisions. Logic was created to then stop the photon at the boundary and project it along the same path using new scattering parameters. There are nine different combinations of old and new positions, three possibilities for the old position and three possibilities for the new position. Each specific situation is investigated at each collision and upon boundary crossing the correct stopping formula is applied.

APPENDIX D

I. AUTOMATION OF RELATIVE FLUX CONTOUR PLOTTING IN DRLITE ROUTINE

A. INTRODUCTION

One very important output of the Monte Carlo routine is the relative flux per unit area at numerous grid points within the reference spheres with respect to the original energy projected by the point source. Knowing the relative flux at numerous grid points allows generation of equal photon flux contours. Calculation of the contours was a tedious job requiring many hours on a programmable pocket calculator and at the drawing board. Accuracy of the contours suffered (not to mention the one doing the work) therefore a method was created to produce many of the contour plots found within this report.

B. AUTOMATION

A subroutine, CONISD, was found in the computer program library that was designed to produce a contour map of irregularly spread data points. Each data point is a triad of x, y and z values where

$$z = f(x,y)$$

This procedure is nicely suited for plotting the flux contours required since the flux per unit area has been calculated at

each increment in range and scattering angle. In transforming evenly spaced range and angle intervals from polar to cartesian coordinates, irregularly spaced intervals are formed. By inspection the flux varies quite linearly in the range direction.

The region to be contoured is subdivided into triangular areas, using acute triangles as much as possible. The method of triangularization is based on that of Ref. 30. The list of triangles is then analyzed for adjacencies. For a given contour level, each outer boundary of the area to be contoured is checked for a possible entry value. Upon finding a value, the contour line is traced from triangle to triangle until it exits the area. As the line is traced from segment to segment, the four nearest values are linearly interpolated and the resulting value stored. After all lines have been found and intersections stored, another subroutine is called to fit smooth curves to the stored values. The resulting curves are the desired contours of equal photon flux.

C. ADDITIONS TO DRLITE

A section was created within DRLITE to drive the subroutine CONISD after computation of relative flux at desired grid points. Inputs required by CONISD are the number of grid points, each corresponding triad of values, the value of each contour desired, scaling factors for plotting the output and designation of which data points are to be boundary points.

Because data used to generate the contours was generated by a Monte Carlo routine, a measure of statistical significance had

to be included before actual plotting. The measure used was based purely on the number of photon crossings at each particular angular bin at every range. If the number of crossings was less than three, the corresponding data triad was removed from the list to be used in generation of contours.

D. DISCUSSION

The automation method explained above, although useful in reducing manual labor, is quite consumptive of CPU time. Computer time goes up almost expotentially with the number of grip points used in the routine. This can be attributed to the number of searches necessary in passing from segment to segment. Although the general idea behind automation of the contour plotting is good, the actual implementation of such methods must be done with caution.

As an aside, the subroutine CONISD also has the capability to have internal cut out boundaries placed into its logic. Future application of this feature may be to draw equal photon flux contours for light propagation around defined objects.

APPENDIX E

I. DERIVATION, DOCUMENTATION AND VERIFICATION OF EFFECTIVE ATTENUATION COEFFICIENT METHOD

A. INTRODUCTION

Gordon [9] presents the concept of an effective attenuation coefficient (EAC) which considerably simplifies multiple scattering calculations. Concise expressions for the total flux through an aperture and the beam-spread function are derived in terms of the EAC. A program was written to evaluate these expressions so that comparison with other light scattering models could be accomplished. This Appendix derives the expressions given by Gordon, documents the routine written to evaluate the expressions and shows verification of program accuracy.

B. DERIVATION

Gordon first suspected that simple formulas might describe multiple scattering after examining Duntley's [29] measurements of the fraction of power emitted from a collimated source which reaches a circular aperture subtending a cone of half-angle θ when viewed from the source and located a distance r from it. Gordon noticed that on semi-log paper the data formed straight lines to about 15 extinction lengths when plotted as a function of range. The expression for flux reaching the aperture where F_o is the source strength can be expressed as [9]:

$$F(\theta,R) = F_o e^{-\alpha_e(\theta)R}, \tag{1}$$

where $\alpha_e(\theta)$ is the EAC as a function of θ.

Consider the geometry of Figure 38. A unidirectional point source is centrally located on the plane x = 0. On the plane x = R is located a circular aperture which subtends a cone of half-angle θ with the point source. The total flux through the aperture is given by [9]:

$$F = F_o \exp\left[-\left(1-\frac{s}{\alpha}\int_0^1 \left\{\int_0^{2\pi}\int_0^{\theta'} P(\theta)d\omega\right\} dt\right)\alpha R\right] \tag{2}$$

where,

$d\omega = \sin\theta \, d\theta \, d\phi$ = differential solid angle

s = scattering coefficient in km^{-1}

α = extinction coefficient in kn^{-1}

t = x/R = normalized range

$$\theta' = \tan^{-1}\left(\frac{\tan\theta}{1-x/R}\right)$$

and $P(\theta)$ is the normalized phase function, thus it satisfies the relation,

$$\int_0^{2\pi}\int_0^{\pi} P(\theta)\, d\omega = 1 \tag{3}$$

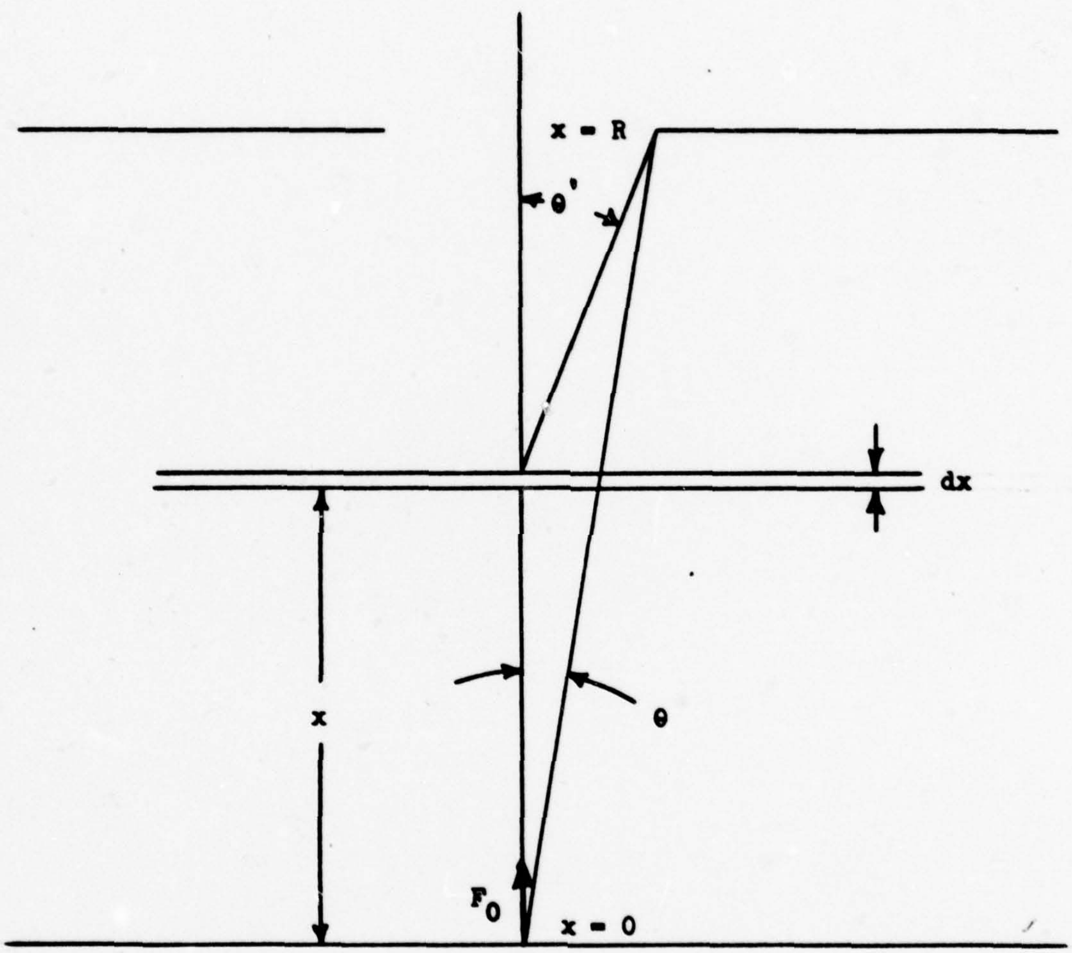


Figure 38

Geometry of EAC Method

A simple physical description of equation 2 is that it accounts for flux absorbed and scattered from the beam in the first term of the exponent and also accounts for flux that is scattered back through the aperture in the integral term of the exponent.

Comparing equation 1 with equation 2 the effective attenuation coefficient is found to be

$$\alpha_e(\theta) = \alpha\left(1-\frac{s}{\alpha}\int_0^1\left\{\int_0^{2\pi}\int_0^{\theta'} P(\theta)d\omega\right\}dt\right) \qquad (4)$$

In adapting this equation to machine computation, a normalized phase function is necessary. A reasonable choice is the Henyey-Greenstein phase function [7],

$$P(\theta) = \frac{(1-G^2)}{4\pi(1+G^2-2G\cos\theta)^{1.5}} \qquad -1<G<1 \qquad (5)$$

Because the Monte Carlo model makes use of this phase function also, comparison of the two models is possible.

Now that the flux through an aperture is known, it can be used to derive the irradiance caused by a unit strength unidirectional point source at each point of a selected target plane. This irradiance distribution is called the beam spread function, $H_B(\theta,R)$. Let the initial point source have unit strength. By considering the flux through a small annulus of width R $d\theta$ located at (R,θ), $H_B(\theta,R)$ can be written as [9],

$$H_B(\theta,R) = \frac{1}{2\pi R^2 \sin\theta} \frac{dF}{d\theta} \qquad (6)$$

where $dF/d\theta$ is the derivative of equation 2 with $F_o = 1$. To evaluate this derivative, Leibnitz rule is used [33]. It states that if

$$F(t) = \int_{a(t)}^{b(t)} \emptyset(x,t)dt \qquad (7)$$

where a(t) and b(t) are differentiable functions of t and where $\emptyset(x,t)$ and $\frac{\partial\emptyset}{\partial t}$ are continuous in x and t, then

$$\frac{dF}{dt} = \int_{a(t)}^{b(t)} \frac{\partial\emptyset(x,t)}{\partial t} dx + \emptyset[b(t),t]\frac{db(t)}{dt} - \emptyset[a(t),t]\frac{da(t)}{dt} \qquad (8)$$

Take the derivative of equation 2,

$$\frac{dF}{d\theta} = F\frac{d}{d\theta}\left[-\left(1-\frac{s}{a}\int_0^1\left\{\int_0^{2\pi}\int_0^{\theta'} P(\theta)d\omega\right\}dt\right)\partial R\right] \qquad (9)$$

$$= 2\pi FsR \frac{d}{d\theta}\int_0^1\left[\int_0^{\theta'} P(\theta)\sin\theta\, d\theta\right]dt \qquad (10)$$

and apply Leibnitz rule where

$$\emptyset(t,\theta) = \int_0^{\theta'} P(\theta)\sin\theta\, d\theta \qquad \begin{array}{l} a(t) = 0 \\ b(t) = 1 \end{array} \qquad (11)$$

it is now apparent that the derivative becomes,

$$\frac{dF}{d\theta} = 2\pi FsR \int_0^1 \left(\frac{\partial}{\partial\theta} \int_0^{\theta'} P(\theta)\ \sin\theta\ d\theta \right) dt \qquad (12)$$

The derivative that remains is of an integral whose range is over the same independent variable, θ, therefore the result is the product of the integrand evaluated at the upper limit and the derivative of the upper limit, i.e.,

$$\frac{\partial}{\partial\theta} \int_0^{\theta'} P(\theta)\ \sin\theta d\theta = P(\theta')\ \sin(\theta')\ \frac{\partial}{\partial\theta}(\theta') \qquad (13)$$

but

$$\frac{\partial}{\partial\theta} \left[\tan^{-1} \left(\frac{\tan\theta}{1-t} \right) \right] = \frac{1}{\left[1 + \left(\frac{\tan\theta}{1-t} \right)^2 \right] (1-t)\cos^2\theta} \qquad (14)$$

so it is clear that

$$\frac{dF}{d\theta} = 2\pi FsR \int_0^1 P(\theta')\sin(\theta')\ \frac{1}{\left[1 + \left(\frac{\tan\theta}{1-t} \right)^2 \right] (1-t)\cos^2\theta}\ dt \qquad (15)$$

Using the small angle approximation, $\tan\theta = \theta = \sin\theta$ and letting

$$u = \tan^{-1} \left(\frac{\theta}{1-t} \right) \qquad (16)$$

$$du = \frac{dt}{\left[1 + \left(\frac{\theta}{1-t} \right)^2 \right] (1-t)} \qquad (17)$$

equation 15 takes the form

$$\frac{dF}{d\theta} = 2\pi FsR\int_0^{\pi/2} P(u)\cos(u)\,du \qquad (18)$$

Using equation 18 in equation 6 the beam spread function is

$$H_B(\theta,R) = \frac{sF}{R\theta}\int_0^{\pi/2} P(u)\cos u\,du \qquad (19)$$

A useful calculation in comparing this beam spread function with that found using the Monte Carlo method is taking the relative logarithm of the beam spread function.

C. ADAPTATION OF EFFECTIVE ATTENTUATION COEFFICIENT METHOD TO MACHINE COMPUTATION

The Effective Attentuation Coefficient method lends itself nicely to machine computation. The inputs required by the program are the scattering coefficient, the extinction coefficient and the Henyey-Greenstein G parameter. Also entered are the initial range and theta, the increment in range and theta and the number of grid points desired in each dimension.

The routine written is called EAC. EAC calls on a library subroutine DQG32 which integrates functions FCC and FCT inside of which the integrands of equation 4 and 19 are defined, respectively. DQG32 uses 32 point Gaussian quadrature which integrates polynomials up to degree 63 exactly.

The output of EAC can be received by numerical matrix tabulation or by graphical contour plotting as described in Appendix D. Examples of these plots are found in Figures 6 through 9.

D. VERIFICATION OF EAC

The results from EAC, the NPS routine, were compared to figures obtained by Gordon [9] using the same method. There is a small dependence on phase function for small optical thicknesses for which this method is applicable. Thus in comparing results with Gordon, a slight deviation is noticed due to the fact that the phase function used may not have been exactly the same. A Henyey-Greenstein G parameter of .94 was used to generate the figures presented here. Figure 39 compares the results of EAC with Gordon for irradiance as a function of θ at a range of 2.12 extinction lengths. Agreement is seen to be quite good. Figure 40 compares the results of EAC with Gordon for irradiance as a function of range with a scatter angle of 10 degrees. Agreement is good here also. These are a few of many checks on routine accuracy which lead to much confidence in the routine.

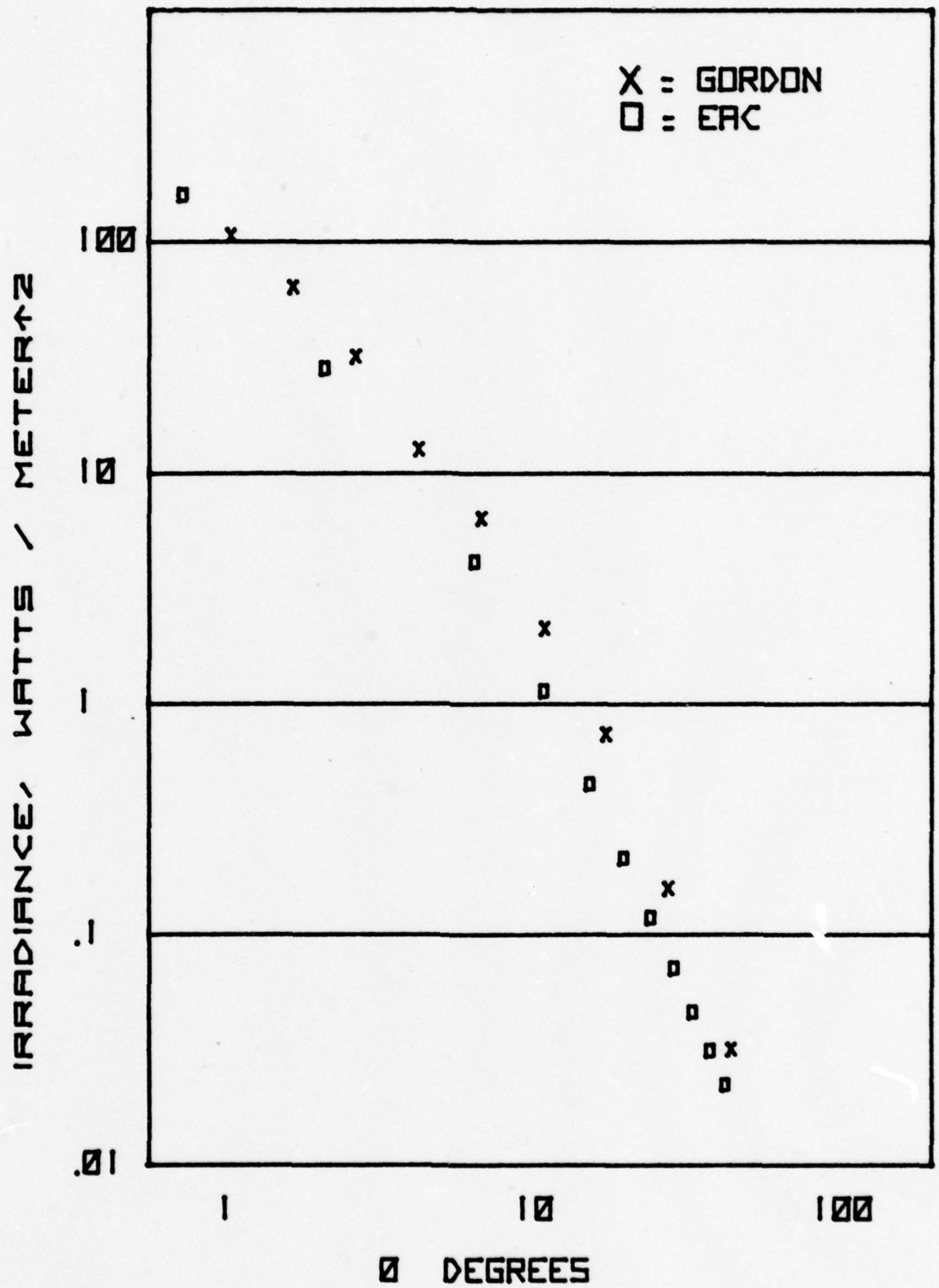


Figure 39

Verification of EAC Method

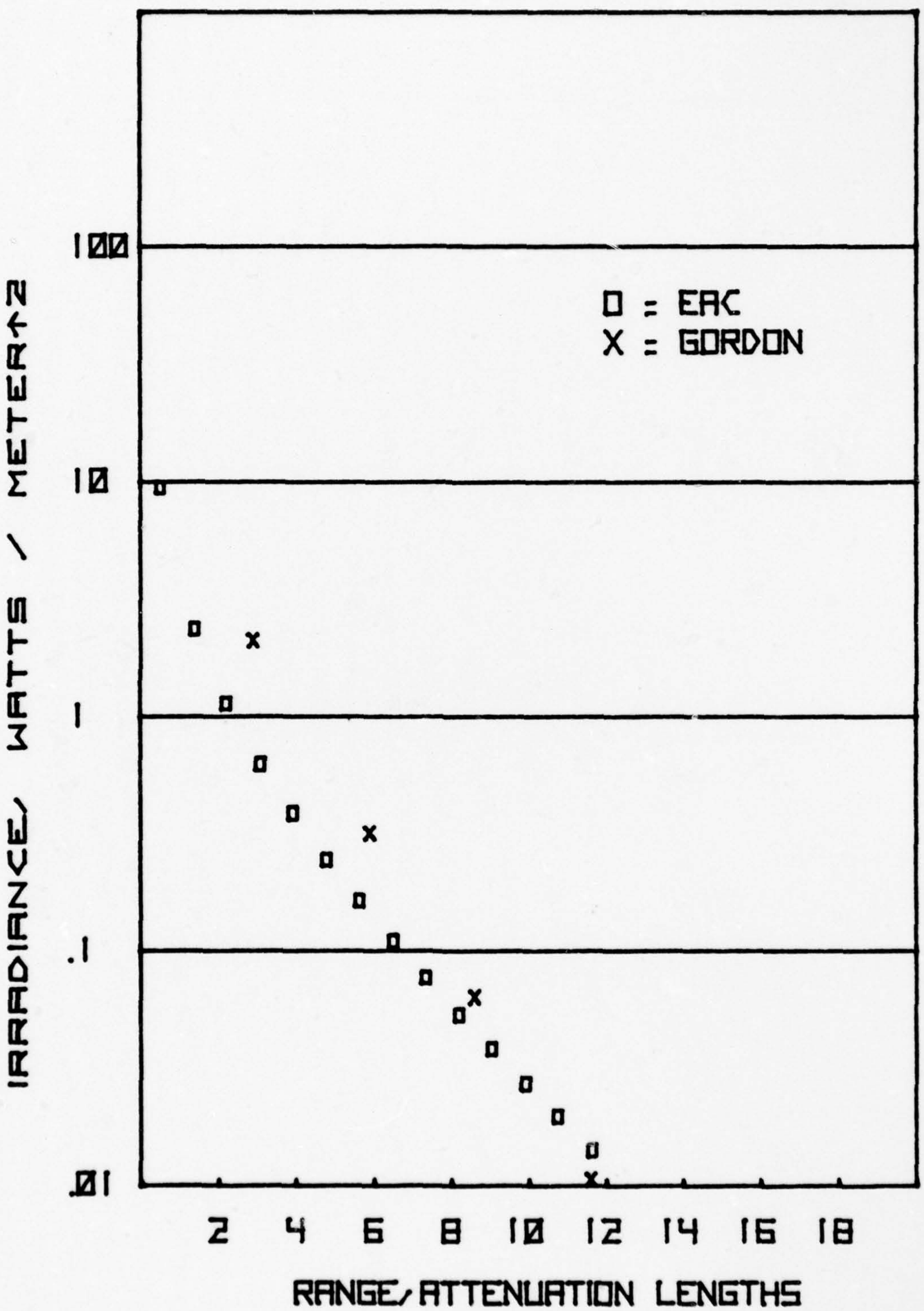


Figure 40

Verification of EAC Method

APPENDIX F

I. CHECKS ON POSSIBLE ERRORS

A. GENERAL

Each computational procedure employed in this simulation was verified when possible and every effort was made to ensure the entire simulation functioned properly. Results of various theories were compared using identical parameters to build confidence at intermediate steps along the path toward final results. Procedures used to document the correct behavior of the computer routines used in this simulation are described in this Appendix.

B. SPECIFIC CHECKS

One of the secondary objectives of this report was to compare results of different theories in predicting light transfer through a scattering absorbing medium. The ability to compare results of separate theories stands by itself as a verification of the accuracy of each theory.

Reference 6 states numerous initial methods used to verify the mechanics of the computer simulation in its first stages. In converting the routine from one that simulates an infinite homogeneous medium to one that contains a cloud, additional checks were required. Many photons were traced manually by hand calculations to ensure that all nine possible combinations of transfer across boundaries were correctly handled by the

routine. Errors were found and corrected on many an occasion. In the situation where the transmission parameters and the phase function of the cloud were the same as those of the surrounding medium, one would expect the results to be identical to those found when simulating a homogeneous medium. This in fact was the case. Results were identical to well within statistical scatter.

Changeover from measurements in terms of extinction lengths to measurements in terms of real dimensions also required brief checking. One would expect the results to be identical when comparing a medium with an extinction coefficient of one inverse kilometer to a medium described in terms of optical thickness. This is so because one unit of distance is the same in both cases, one kilometer. This is the case. In comparing trials where the first has a distance between accountability shells of twice the second trial but an extinction coefficient of half the second trial with the albedo the same, the number of photons crossing each bin should be the same. Once again this is the case.

In some cases the results were checked by intuition only as no previous work with similar presentation was found. Specifically, the equal photon flux contours when passing through clouds were checked only by what one would expect them to look like.

Many test runs were made using the model generating a scatter angle weighted by an arbitrary phase function. The

angles generated are in fact weighted by areas of panels and linearly within, panels.

Confidence in the statistical nature of a Monte Carlo routine is related to the number of photon histories used in each trial. The smallest number of photons possible were used to create outputs of the desired accuracy in order to minimize computer time consumption. In most cases of spatial spread thousands of photon histories were generated. In time spread cases as few as 100 were required at times.

Other checks made are found in Ref. 6 as well as within the body of this work.

APPENDIX G

I. DERIVATION OF CLOSED FORM EXPRESSION FOR TIME SPREAD

Stotts [26] derives a closed form expression for time spread as follows. From vector analysis, the average pathlength of a photon per unit time is

$$\frac{dR}{dt} = \left[(\frac{dz}{dt})^2+(\frac{dy}{dt})^2+(\frac{dz}{dt})^2\right]^{\frac{1}{2}}$$

which reduces easily, where $\bar{r} = x\bar{i} = y\bar{j}$, to

$$\frac{dR}{dt} = \left[1+ |\frac{d\bar{r}}{dz}|^2\right]^{\frac{1}{2}} dz$$

From basic scattering theory [31] the projected angle of scattering is approximated by

$$[\omega_o\tau\ \gamma_o^2]^{\frac{1}{2}}$$

where ω_o is the albedo of single scattering, τ is the optical thickness of the scattering region and γ_o is the RMS scatter angle. Thus the projected mean-squared transverse displacement is

$$|d\bar{r}| = z(\omega_o\tau\ \gamma_o^2)^{\frac{1}{2}}$$

thus

$$\left|\frac{d\bar{r}}{dz}\right| = \left(\frac{9}{4}\ \omega_o \tau\ \gamma_o^2\right)^{\frac{1}{2}}$$

and

$$dR = \left[1+\frac{9}{4}\ \omega_o \tau\ \gamma_o^2\right]^{\frac{1}{2}} dz$$

Integrating results in

$$R = \frac{8z}{27\omega_o \tau\ \gamma_o^2}\left[\left(1+\frac{9}{4}\ \omega_o \tau\ \gamma_o^2\right)^{1.5}-1\right]$$

The average multipath time spread is defined as the difference between the average transient time incurred from multiple scattering and the normal transient time in the absence of scattering. Thus, Stotts has for time spread,

$$L = \frac{z}{c}\left[\frac{.30}{\omega_o \tau\ \gamma_o^2}\left[\left(1+\frac{9}{4}\ \omega_o \tau\ \gamma_o^2\right)^{1.5}-1\right]-1\right]$$

```
C**** MAIN ROUTINE OF MIE SCATTERING PROGRAM
C
C     THIS ROUTINE CALCULATES THE MIE SCATTERING FUNCTIONS AVERAGED
C         OVER A PARTICLE DISTRIBUTION FUNCTION OF THE FORM
C         X**ALPHA*EXP(-B*X**GAM)  (SEE DEIRMENDJIAN, ELSEVIER, 1969)
C     INPUT:  NX,NTH,ICODE,RXO,DELX,M,ALPHA,GAM,XC (I3,I2,I1,F4.2,6F5.3)
C         NX = NUMBER OF VALUES OF PARTICLE SIZES TO BE CALCULATED OVER
C         NTH = NUMBER OF VALUES OF THETA AT WHICH TO CALCULATE SCATTERIN
C              INTENSITY EVENLY INCREMENTED FROM 0 TO 180 DEGREES
C         ICODE = 0 FOR MINIMUM OUTPUT
C                GT.0 TO GET SCATTERING AND EXTINCTION PRINTED AT EACH X
C                GT 5 TO GET SCATTERING FUNCTION PRINTED AT EACH X
C         RXO = INITIAL VALUE OF X, REDUCED PARTICLE SIZE: 2*PI*R/LAMBDA
C         DELX = INCREMENT OF PARTICLE SIZE
C         M = INDEX OF REFRACTION (2F5.3), REAL AND IMAGINARY
C         ALPHA = PARAMETER IN DISTRIBUTION FUNCTION (SEE ABOVE)
C         GAM = "
C         XC = MOST PROBABLE VALUE OF X IN DISTRIBUTION FUNCTION
C             NOTE, B = ALPHA/(GAM*XC**GAM)
C     IF NX = 0, PROGRAM STOPS EXECUTION
      COMPLEX M
      REAL *4AS1(90),AS2(90),WEIGHT(501),S1(90),S2(90)
      DATA PI/3.1415926536/
C**** READ IN DATA
   20 READ (5,320) NX,NTH,ICODE,RXO,DELX,M,ALPHA,GAM,XC
      IF (NX.EQ.0) STOP
      WRITE (6,340) NX,NTH,RXO,DELX,M,ALPHA,GAM,XC
      IF ((NX/2)*2.NE.NX) GO TO 40
      WRITE (6,360)
      GO TO 20
C**** INITIALIZE
   40 SAV = (ALPHA+1.0)/GAM
      B = ALPHA/(GAM*XC**GAM)
      ANORM = (GAM*B**SAV)/(GAMMA(SAV))
      RX = RXO-DELX
C**** CALCULATE DIXTRIBUTION FUNCTION (WEIGHT(I))
C
      DO 60 I=1,NX
      RX = RX+DELX
      WEIGHT(I) = ANORM*RX**ALPHA*EXP(-B*RX**GAM)
      IF (WEIGHT(I).GT.1.00E-08) GO TO 60
      IF (RX.LT.XC) GO TO 60
      NX = I
```

```
      GO TO 80
   60 CONTINUE
C
   80 CONTINUE
      IF (NX.EQ.1) WEIGHT(1)=1.00
      WRITE (6,380) (WEIGHT(I),I=1,NX)
      RX = RXC-DELX
      ICOEF = 2
C**** START LOOP FOR EACH VALUE OF X
C
      DO 240 IC=1,NX
      RX = RX+DELX
      CALL MIEM (ICODE,NTH,RX,M,AS1,AS2,QSCA,QEXT)
C**** PRINT OUT FOR EACH VALUE OF X IF DESIRED (ICODE.GE.5)
      IF (ICODE.LT.5) GO TO 100
      WRITE (6,400) (AS1(J),J=1,NTH)
      WRITE (6,400) (AS2(J),J=1,NTH)
  100 IF (ICODE.LT.1) GO TO 120
      WRITE (6,420) RX,QSCA,QEXT
C**** BEGIN INTEGRATION USING SIMPSON'S RULE, FIRST VALUE STARTS HERE
  120 IF (IC.NE.1) GO TO 160
      SWT = WEIGHT(1)
      BETASC = PI*RX**2*QSCA*WEIGHT(IC)
      BETAEX = PI*RX**2*QEXT*WEIGHT(IC)
C
      DO 140 I=1,NTH
      S1(I) = AS1(I)*WEIGHT(IC)
  140 S2(I) = AS2(I)*WEIGHT(IC)
C
      GO TO 240
C**** COMPLETE INTEGRATION USING SIMPSON'S RULE; LAST VALUE ENTERED HERE
  160 IF (IC.NE.NX) GO TO 200
      SWT = (SWT+WEIGHT(IC))*DELX/3.0
      BETAEX = (BETAEX+WEIGHT(IC)*PI*RX**2*QEXT)*DELX/(3.0)
      BETASC = (BETASC+WEIGHT(IC)*PI*RX**2*QSCA)*DELX/(3.0)
C
      DO 180 I=1,NTH
      S1(I) = (S1(I)+WEIGHT(IC)*AS1(I))*DELX/(3.0)
  180 S2(I) = (S2(I)+WEIGHT(IC)*AS2(I))*DELX/(3.0)
C
      GO TO 240
C**** INTERMEDIATE STEPS IN INTEGRATION USING SIMPSON'S RULE
  200 ICOEF = 8/ICOEF
      SWT = SWT+ICOEF*WEIGHT(IC)
      BETASC = BETASC+ICOEF*PI*RX**2*QSCA*WEIGHT(IC)
      BETAEX = BETAEX+ICOEF*PI*RX**2*QEXT*WEIGHT(IC)
C
      DO 220 I=1,NTH
```

```
      S1(I) = S1(I)+ICOEF*WEIGHT(IC)*AS1(I)
  220 S2(I) = S2(I)+ICOEF*WEIGHT(IC)*AS2(I)
C
C**** PRINT OUT S1 EVERY 10TH CALCULATION
      IF ((IC/10)*10.NE.IC) GO TO 240
C     WRITE(6,996) RX,BETASC
C     WRITE(6,997) (S1(I), I = 1,NTH)
  240 CONTINUE
C
C**** COMPLETE LOOP
C**** PRINT OUT RESULTS
      WRITE (6,440) SWT,BETASC,BETAEX
      WRITE (6,460)
      WRITE (6,480) (S1(I),I=1,NTH)
      WRITE (6,480) (S2(I),I=1,NTH)
C
      DO 260 I=1,NTH
      S1(I) = S1(I)/BETASC
      S2(I) = S2(I)/BETASC
  260 CONTINUE
C
      WRITE (6,500)
      WRITE (6,480) (S1(I),I=1,NTH)
      WRITE (6,480) (S2(I),I=1,NTH)
C
      DO 280 I=1,NTH
  280 S1(I) = 0.5*(S1(I)+S2(I))
C
      WRITE (6,520)
      WRITE (6,480) (S1(I),I=1,NTH)
C
C**** THIS PROVIDES PUNCHED CARDS USABLE IN MONTE CARLO ROUTINE
      DO 300 J=1,2
  300 WRITE (7,540) (S1(I),I=1,NTH)
C
      GO TO 20
C
  320 FORMAT (I3,I2,I1,F4.2,6F5.3)
  340 FORMAT ('1',' NX = ',I3,'  NTH = ',I3,'  RXO = ',F10.4,
     1 '  DELX = ',F10.4,'  M = ',2F10.4,/,' ALPHA = ',F12.4,'  GAM = ',
     2 F12.4,' XC = ',F12.4)
  360 FORMAT (' NO. OF POINTS FOR NX MUST BE ODD TO APPLY SIMPSON''S RUL
     1E')
  380 FORMAT (' WEIGHTING FUNCTION:',/,(1X,10F12.6))
  400 FORMAT (1X,10F12.6)
  420 FORMAT (' X = ',F10.4,'   QSCA = ',F12.6,'   CEXT = ',F12.6)
  440 FORMAT (' SWT = ',F12.6,'   BETASC = ',F12.6,'   BETAEX = ',F12.6)
  460 FORMAT (/,' THE AVERAGED SCATTERING FUNCTIONS ARE:',/)
```

```
  480 FORMAT (1X,10F12.5)
  500 FORMAT (/,' VALUES OF P/4PI TO COMPARE WITH DEIRMENDJIAN ARE')
  520 FORMAT (/,' THE AVERAGED NORMALIZEC SCATTERING PHASE FUNCTION IS')
  540 FORMAT (7F10.6)
      END


C**** SUBROUTINE ANGLE CF MIE SCATTERING PROGRAM ****
C
      SUBROUTINE ANGLE (A,B,NS,NTH,AS1,AS2)
      REAL *4L,AS1(90),AS2(90)
      COMPLEX *8A(120),B(120),S1(90),S2(90)
      DATA PI/3.1415926536/
      IF (NS.LT.121) GO TO 20
      WRITE (6,80)
      RETURN
   20 CONTINUE
CP    WRITE(6,999) (A(I), B(I), I = 1,NS)
CP999 FORMAT(1X,3(2F8.4,2X,2F8.4))
      DTHETA = PI/(NTH-1)
      THETA = -DTHETA
C
      DO 60 J=1,NTH
      THETA = THETA+DTHETA
CP    WRITE(6,997) THETA
CP997 FORMAT(' THETA = ',F10.5)
      CT = COS(THETA)
      ST = SIN(THETA)
      P = CT
      PP = 1.0
      PPP = 0.0
      PM1 = 1.0
      PM1P = 0.0
      PM1PP = 0.0
      S1(J) = 1.5*(A(1)+B(1)*CT)
      S2(J) = 1.5*(B(1)+A(1)*CT)
C
      DO 40 N=2,NS
      L = N
      PM2 = PM1
      PM2P = PM1P
      PM2PP = PM1PP
      PM1 = P
      PM1P = PP
      PM1PP = PPP
```

```
      P = (2.0*L-1.0)*CT*PM1/L-(L-1.0)*PM2/L
      PP = ((2.0*L-1.0)*CT*PM1P-(L-1.0)*PM2P+(2.0*L-1.0)*PM1)/L
      PPP = ((2.0*L-1.0)*CT*PM1PP-(L-1.0)*PM2PP+(4.0*L-2.0)*PM1P)/L
      C = (2.0*L+1.0)/(L*(L+1.0))
      SC = CT*PP-ST**2*PPP
      S1(J) = S1(J)+(A(N)*PP+B(N)*SC)*C
      S2(J) = S2(J)+(B(N)*PP+A(N)*SC)*C
   40 CONTINUE
C
CP    WRITE(6,998) S1(J), S2(J)
CP998 FORMAT(5X,2(2F8.4,5X))
      AS1(J) = CABS(S1(J))**2
      AS2(J) = CABS(S2(J))**2
   60 CONTINUE
C
CP    WRITE(6,110) (AS1(J), J = 1,NTH)
CP    WRITE(6,110) (AS2(J), J = 1,NTH)
CP110 FORMAT(1X,10F12.6)
      RETURN
C
   80 FORMAT (' *** ERROR WILL OCCUR IN ANGLE DUE TO IMPROPER INDEXING')
      END


C**** SUBROUTINE MIEM OF MIE SCATTERING PROGRAM ****
C
C THE FOLLOWING FUNCTIONS MUST BE INCLUDED
C     JN(N,X) CALCULATES PSI, RELATED TO BESSEL FUNCTION
C     JNP(N,X) CALCULATED DERIVATIVE OF PSI
C     FN(N,X) CALCULATES CHI, RELATED TO NEUMAN FUNCTION
C     FNP(N,X) CALCULATED DERIVATIVE OF CHI
C     HN(N,X) CALCULATES XI, INVOLVING HAENKEL FUNCTION
C     HNP(N,X) CALCULATES DERIVATIVE OF XI
C     CSIN(X) CALCULATES THE SIN OF X,, X MAY BE COMPLEX
C     CCOS(X) CALCULATES COSINE OF X,, X MAY BE COMPLEX
C   INPUT CONSISTS OF
C      N,NTH,RX,EP  (I1,I3,1X,3F5.2)
C     N = 0 TO START A NEW PAGE, = 1 IF NO NEW PAGE TO START
C      NTH = NUMBER OF VALUES OF THETA BETWEEN 0 AND PI
C      RX = X (REDUCED PARTICLE SIZE IN NOTATION OF VAN DE HULST)
C      EP = DIELECTRIC CONSTANT = SQUARE OF INDEX OF REFRACTION
C
C     M IS THE COMPLEX REFRACTIVE INDEX
C   1 READ(5,100) N,NTH,RX,M
C 100 FORMAT(I1,I3,1X,6F5.2)
```

```
      SUBROUTINE MIEM (ICODE,NTH,RX,M,AS1,AS2,QSCA,QEXT)
      COMPLEX X,JN,FN,HN,JNP,FNP,HNP,M,Y
      COMPLEX JX(3),JY(3),HX(3),JPX, JPY, HPX, AN , BN
      COMPLEX MU,EP,RMU,REP
      COMPLEX *8 A(120),B(120)
      REAL *4AS1(90),AS2(90)
      N = 1
      X = RX
      DEL = 0.0
      MU = (1.0,0.0)
      EP = M**2
      RMU = CSQRT(MU)
      REP =  CSQRT(EP)
      IF (NTH.LE.0) STOP
      N = 1-N
      Y = M*X
      EPSIL = 1.0E-6*X**2/(2*(2*N+1))
      QEXT = 0.0
      QSCA = 0.0
C
C SET UP FIRST THREE FUNCTIONS OF EACH KIND REQUIRED
C
      JX(1) = JN(0,X)
      JX(2) = JN(1,X)
      JX(3) = JN(2,X)
      JY(1) = JN(0,Y)
      JY(2) = JN(1,Y)
      JY(3) = JN(2,Y)
      HX(1) = HN(0,X)
      HX(2) = HN(1,X)
      HX(3) = HN(2,X)
C
      DO 20 N=1,120
      JPX = (N+1)*JX(2)/X-JX(3)
      JPY = (N+1)*JY(2)/Y-JY(3)
      HPX = (N+1)*HX(2)/X-HX(3)
C
C CALCULATE THE COEFFICIENTS OF THE SCATTERED WAVE FUNCTION
C
      AN = (RMU*JPY*JX(2)-REP*JY(2)*JPX)/(RMU*JPY*HX(2)-REP*JY(2)*HPX)
      BN = (REP*JPY*JX(2)-RMU*JY(2)*JPX)/(REP*JPY*HX(2)-RMU*JY(2)*HPX)
      A(N) = AN
      B(N) = BN
      DELQ = (2*N+1)*(AN+BN)
      DELQS = (2*N+1)*(CABS(AN)**2+CABS(BN)**2)
      QEXT = QEXT+DELQ
      QSCA = QSCA+DELQS
C
```

```
C CONVERGENCE CRITERION TESTED HERE
C
      IF (ABS(DELQ).LE.EPSIL) GO TO 40
C
C     CALCULATE THE NEXT HIGHER ORDER FUNCTION FOR NEXT TERM IN SERIES
C
      JX(1) = JX(2)
      JX(2) = JX(3)
      JY(1) = JY(2)
      JY(2) = JY(3)
      HX(1) = HX(2)
      HX(2) = HX(3)
      JX(3) = (2*N+3)*JX(2)/X-JX(1)
      JY(3) = (2*N+3)*JY(2)/Y-JY(1)
      HX(3) = (2*N+3)*HX(2)/X-HX(1)
   20 CONTINUE
C
      WRITE (6,60) N,DELQ,X
   40 CONTINUE
      NS = N
      CALL ANGLE (A,B,NS,NTH,AS1,AS2)
      QEXT = QEXT*2.0/X**2
      QSCA = QSCA*2.0/X**2
      RETURN
C
   60 FORMAT (/,I4,' TERMS WERE USED TO OBTAIN CONVERGENCE.  LAST TERM W
     1AS',G12.3,' AT X = ',2G12.3)
      END


C**** COMPLEX FUNCTIONS JNP, HN, FN, JN, CSIN, CCOS OF MIE SCAT PROGRAM
C
      COMPLEX FUNCTIONJNP(N,X)
      COMPLEX X,JN
      N1 = N+1
      JNP = (N+1.0)*JN(N,X)/X-JN(N1,X)
      RETURN
      END
C     FLOWCHART FOR FNP
      COMPLEX FUNCTIONFNP(N,X)
      COMPLEX X,FN
      N1 = N+1
      FNP = (N+1.0)*FN(N,X)/X-FN(N1,X)
      RETURN
      END
```

```
      COMPLEX FUNCTIONHN(N,X)
      COMPLEX X,JN,FN
      HN = JN(N,X)+(0.0,1.0)*FN(N,X)
      RETURN
      END
      COMPLEX FUNCTIONHNP(N,X)
      COMPLEX X,JNP,FNP
      HNP = JNP(N,X)+(0.0,1.0)*FNP(N,X)
      RETURN
      END
      COMPLEX FUNCTIONFN(N,X)
      COMPLEX X,CCOS,CSIN,FO,F1
      IF (CABS(X).GE.1.0E-9) GO TO 20
      FN = 0.0
      RETURN
   20 FO = -CCOS(X)/X
      IF (N.NE.0) GO TO 40
      FN = -FO*X
      RETURN
   40 F1 = (FO-CSIN(X))/X
      IF (N.NE.1) GO TO 60
      FN = -F1*X
      RETURN
C
   60 DO 80 I=2,N
      FN = (2.0*I-1.0)*F1/X-FO
      FO = F1
      F1 = FN
   80 CONTINUE
C
      FN = -FN*X
      RETURN
      END
      COMPLEX FUNCTIONJN(N,X)
      COMPLEX X,CCOS,CSIN,JO,J1
      IF (CABS(X).GE.1.0E-9) GO TO 40
      JO = (1.0,0.0)
      IF (N.NE.0) GO TO 20
      JN = (1.0,0.0)
      RETURN
   20 JN = (0.0,0.0)
      RETURN
   40 JO = CSIN(X)/X
      IF (N.NE.0) GO TO 60
      JN = JO*X
      RETURN
   60 J1 = (JO-CCOS(X))/X
      IF (N.NE.1) GO TO 80
```

```
      JN = J1*X
      RETURN
C
   80 DC 100 I=2,N
      JN = (2.0*I-1.0)*J1/X-J0
      J0 = J1
      J1 = JN
  100 CONTINUE
C
      JN = JN*X
      RETURN
      END
      CCMPLEX FUNCTIONCSIN(X)
      CCMPLEX X
      A = X
      B = AIMAG(X)
      CSIN = SIN(A)*COSH(B)+(0.0,1.0)*COS(A)*SINH(B )
      RETURN
      END
      CCMPLEX FUNCTIONCCOS(X)
      CCMPLEX X
      A = X
      B = AIMAG(X)
      CCCS = COS (A)*COSH(B)-(0.0,1.0)*SIN(A)*SINH(B)
      RETURN
      END



C**** MCNTE CARLO DRIVER ROUTINE DRLITE ****
C
C     THIS ROUTINE IS INTENDED TO CRIVE THE MONTE CARLO ROUTINE
C     CALLED LITE, WHICH CALCULATES THE DISTRIBUTION OF PHOTONS
C     FROM A UNIDIRECTICNAL LIGHT SOURCE.  IT IS USED TO READ
C     INFORMATION INTO THE PROGRAM AND TO CONTROL THE OUTPUT.
C     THE DISTRIBUTION OF PHOTONS AND THE TOTAL NUMBER OF PHOTONS
C     IN EACH SHELL ARE STANDARD OUTPUT INFORMATICN.
C     CPTIONAL OUTPUT IS CONTROLLED BY METHOD AND IPRT STATE-
C     MENTS.  METHCD CONTROLS THE SHAPE OF THE THETA BINS,
C     WHILE IPRT CONTROLS THE ACTUAL OUTPUT.  PLOT INDICATES IF
C     PLOT CF SPATIAL INFORMATION IS DESIRED.
      REAL *8GIN
      DIMENSION G(2), SCA(2) , ABB(2), N(2), RPT(2)
      DIMENSION PTIM(80), XVAL(80)
      REAL*4 THTA(20),X(200),Y(200),Z(2CC),C(8),RFLUX(10,20,1)
      REAL*8 TL(12)/'PLCT CF ','NEG LOG ','OF RELAT','IVE FLUX','  M A M
```

```
     *I','LLBACH   ',6*'          '/                                    DRL00090
      REAL *8XAX/' KMETER '/                                            DRL00100
      REAL *8YAX/' KMETER '/                                            DRL00110
      INTEGER *4IB(4),NB(1)                                             DRL00130
      INTEGER NA(10) /1,0,0,0,1,0,1,1,0,0/                              DRL00140
      INTEGER*4 BINS(10,20,1),BINDST(10,20,20)                          DRL00150
      COMMON WEIGHT(2,70),PHASE(2,70),SLOPE(2,70),THETA(2,70)           DRL00160
      EQUIVALENCE (BINS,RFLUX)                                          DRL00170
      DATA PI/3.1415926536/                                             DRL00180
      CALL ERRSET (209,100,-1,1,1,208)                                  DRL00190
   20 READ (5,560) NPHOT,NTHETA,NFLDVW,NSHLS,N(1),N(2),PLOT,IPRT,METHOD,DRL00200
     1IX                                                                DRL00210
      READ (5,580) DISTSH,THKNES,SCA(1),SCA(2),ABB(1),ABB(2),G(1),G(2),RDRL00220
     1PT(1),RPT(2),FBACK,GIN                                            DRL00230
      IF (METHOD.EQ.0) METHOD=1                                         DRL00240
      IF (NPHOT.EQ.0) STOP                                              DRL00250
      IF (N(1).EQ.0.0) GO TO 100                                        DRL00260
C                                                                       DRL00270
      DO 80 J=1,2                                                       DRL00280
      READ (5,600) (THETA(J,I),I=1,70)                                  DRL00290
      READ (5,600) (PHASE(J,I),I=1,70)                                  DRL00300
      TNORM = 0.0                                                       DRL00310
      TWT = 0.0                                                         DRL00320
      PHASE(J,1) = 0.0                                                  DRL00330
      M = N(J)-1                                                        DRL00340
C                                                                       DRL00350
      DO 40 I=1,M                                                       DRL00360
      PHASE(J,I+1) = PHASE(J,I+1)*SIN(THETA(J,I+1)*PI/180.0)            DRL00370
      PAVG = (PHASE(J,I)+PHASE(J,I+1))/2.0                              DRL00380
      TDIF = (THETA(J,I+1)-THETA(J,I))*PI/180.0                         DRL00390
   40 TNORM = TNORM+PAVG*TDIF                                           DRL00400
C                                                                       DRL00410
C                                                                       DRL00420
      DO 60 I=1,M                                                       DRL00430
      PAVG = (PHASE(J,I)+PHASE(J,I+1))/2.0                              DRL00440
      TDIF = (THETA(J,I+1)-THETA(J,I))*PI/180.0                         DRL00450
      WEIGHT(J,I) = PAVG*TDIF/TNORM                                     DRL00460
      SLOPE(J,I) = ((PHASE(J,I+1)-PHASE(J,I))/(THETA(J,I+1)-THETA(J,I)))DRL00470
     1*180.0/PI                                                         DRL00480
   60 TWT = TWT+WEIGHT(J,I)                                             DRL00490
C                                                                       DRL00500
   80 CONTINUE                                                          DRL00510
C                                                                       DRL00520
  100 WRITE (6,620)                                                     DRL00530
      CALL LITE (NPHOT,NTHETA,NFLDVW,NSHLS,DISTSH,THKNES,SCA,ABB,G,RPT,FDRL00540
     1BACK,IX,BINS,BINDST,METHOD,IPRT,N,PTIM,SIGTIM,AVGTIM)             DRL00550
      WRITE (6,640)                                                     DRL00560
C                                                                       DRL00570
```

```
      DO 240 I=1,NSHLS
      ISUM = 0
C
      DO 120 J=1,NTHETA
C
      DO 120 K=1,NFLDVW
      ISUM = ISUM+BINS(I,J,K)
  120 CONTINUE
C
      WRITE (6,660)
      WRITE (6,680) I,ISUM
      IF (NFLDVW.EQ.1) GO TO 180
C
      DO 160 J=1,NTHETA
      ISUM = 0
C
      DO 140 K=1,NFLDVW
  140 ISUM = ISUM+BINS(I,J,K)
C
      WRITE (6,700) J,ISUM
      WRITE (6,840) (BINS(I,J,K),K=1,NFLDVW)
  160 CONTINUE
C
      GO TO 200
  180 WRITE (6,840) ((BINS(I,J,K),K=1,NFLDVW),J=1,NTHETA)
  200 CONTINUE
      IF(IPRT.LE.3) GO TO 240
      WRITE (6,720)
C
      DO 220 J=1,NTHETA
  220 WRITE (6,740) (BINDST(I,J,K),K=1,20)
C
  240 CONTINUE
C
      IF (IPRT.NE.3) GO TO 280
C
      WRITE (6,780)
      WRITE(6,890)
      WRITE(6,900) PTIM
      WRITE (6,760) THKNES,AVGTIM,SIGTIM
      WRITE(6,920)
      WRITE (6,780)
      DO 260 I=1,80
  260 XVAL(I) = -9.0750+I*.0750
C
      MAX=0.0
C
      DO 270 J=2,80
```

```
  270 IF(PTIM(J).GT.MAX) MAX=PTIM(J)
C
C
      DC 275 I=1,80
  275 PTIM(I)=PTIM(I)/MAX
C
      CALL PLOTP(XVAL,PTIM,80,8)
  280 CCNTINUE
      IF (IPRT.GE.6) GO TO 380
      IF (IPRT.LT.1) GO TC 20
      WRITE (6,800)
C
      DO 320 I=1,NSHLS
      R = DISTSH*I
C
      DO 300 J=1,NTHETA
      TH1 = (PI*(J-1))/NTHETA
      IF (METHOD.EQ.3) TH1=(TH1*(J-1))/NTHETA
      TH2 = (PI*J)/NTHETA
      IF (METHOD.EQ.3) TH2=(TH2*J)/NTHETA
      AREA = 2.0*PI*R**2*(COS(TH1)-COS(TH2))
      THTA(J) = (TH2+TH1)/2.0
C
      DO 300 K=1,NFLDVW
      IF (BINS(I,J,K).LT.3) RFLUX(I,J,K)=0.0
      IF (BINS(I,J,K).LT.3) GO TO 300
      RFLUX(I,J,K) = -ALOG10(BINS(I,J,K)/(AREA*NPHCT))
  300 CCNTINUE
C
      WRITE (6,820) ((RFLUX(I,J,K),K=1,NFLDVW),J=1,NTHETA)
  320 CONTINUE
  380 CONTINUE
  42C IF (PLOT.EQ.0.0) GO TO 20
C*****THIS SECTION IS USED TO DRIVE THE PLOTTING SUBROUTINE CONISD
C
      DC 540 K=1,NFLDVW
      INC = 4
C
      DC 460 I=1,NTHETA
C
      DO 440 J=1,NSHLS
      IF (RFLUX(J,I,K).EQ.0.0) GO TO 440
      INC = INC+1
      X(INC) = J*DISTSH*COS(THTA(I))
      Y(INC) = J*DISTSH*SIN(THTA(I))
      Z(INC) = RFLUX(J,I,K)
  440 CONTINUE
C
```

```
  460 CONTINUE                                                          DRL01530
C                                                                       DRL01540
C                                                                       DRL01550
      X(1) = -DISTSH*NSHLS-1.0                                          DRL01560
      X(2) = -DISTSH*NSHLS-1.0                                          DRL01570
      X(3) = +DISTSH*NSHLS+1.0                                          DRL01580
      X(4) = +DISTSH*NSHLS+1.0                                          DRL01590
      Y(1) = 0.0                                                        DRL01610
      Y(2) = +DISTSH*NSHLS+1.0                                          DRL01620
      Y(3) = +DISTSH*NSHLS+1.0                                          DRL01630
      Y(4) = 0.0                                                        DRL01640
      Z(1) = 500.0                                                      DRL01660
      Z(2) = 500.0                                                      DRL01670
      Z(3) = 500.0                                                      DRL01680
      Z(4) = 500.0                                                      DRL01690
      NPTS=INC                                                          DRL01710
C                                                                       DRL01720
      DO 500 ICORN=1,4                                                  DRL01730
      IB(ICORN) = ICORN                                                 DRL01740
  500 CONTINUE                                                          DRL01750
C                                                                       DRL01760
      NF = 1                                                            DRL01770
      NB(1) = 4                                                         DRL01780
      NC = 8                                                            DRL01790
C                                                                       DRL01800
      DO 520 I=1,NC                                                     DRL01810
      C(I) = -3.0+I*1.0                                                 DRL01820
  520 CONTINUE                                                          DRL01830
C                                                                       DRL01840
      CF = .6                                                           DRL01850
      XS = 9.0                                                          DRL01860
      YS = 4.5                                                          DRL01870
      CALL CONISD (X,Y,Z,NPTS,IB,NR,NB,C,NC,CF,XS,YS,TL,XAX,YAX,NA,IER) DRL01880
      WRITE (6,880) IER                                                 DRL01890
  540 CONTINUE                                                          DRL01900
C                                                                       DRL01910
      GO TO 20                                                          DRL01920
C                                                                       DRL01930
  560 FORMAT (I6,5I2,3I1,I10)                                           DRL01940
  580 FORMAT (11F5.3,D5.3)
  600 FORMAT (7F10.6)                                                   DRL01960
  620 FORMAT (1H1)                                                      DRL01970
  640 FORMAT (/' DISTRIBUTION OF PHOTONS IS GIVEN BY')                  DRL01980
  660 FORMAT (1X)                                                       DRL01990
  680 FORMAT (1X,'TOTAL NUMBER IN SHELL ',I2,' IS ',I6)                 DRL02000
  700 FORMAT ('        NO. IN BIN ',I2,' IS',I6)                        DRL02010
  720 FORMAT (/' TIME OF ARRIVAL BINS GIVEN BY',/)                      DRL02020
  740 FORMAT (1X,20I6)                                                  DRL02030
```

```
760 FORMAT (/,' CLOUD THICKNESS=',F6.3,5X,'AVG TIME=',E12.3,5X,'DEVIATDRL02040
   *ICN=',E12.3)                                                        DRL02050
780 FORMAT (1H1)                                                        DRLC2C6C
800 FORMAT (//,' THE NEGATIVE LOG OF RELATIVE FLUX AT EACH ANULAR RINGDRL02070
   1  IS')                                                              DRL02080
820 FCRMAT (1X,20F6.2)                                                  DRL02C9C
840 FCRMAT (1X,20I6)                                                    DRL02100
860 FORMAT (10E12.3)                                                    DRLC211C
88C FCRMAT (/,' ERROR IER RETURNED=',I5,/)                              DRL02120
890 FCRMAT (' TIME BINS AT EDGE OF CLOUD ARE')                          DRL0213C
900 FORMAT (/,10F12.5)                                                  DRL02140
920 FORMAT(////////////////////////)                                    DRL02145
    ENC                                                                 DRL02150


C**** MCNTE CARLO ROUTINE LITE ****
C
C     THIS SUBROUTINE SIMULATES A PHOTCN WHICH RANCOMLY COLLIDES WITH   LIT00010
C     PARTICLES, SCATTERED AT ANGLES WEIGHTED BY VARIOUS FUNCTIONS, AND LIT00020
C     DETERMINES LOCATION OF INTERSECTICN WITH VARIOUS SPHERES          LIT00030
C                                                                       LIT00040
C        INPUT PARAMETERS:                                              LIT00050
C     NPHOT = NUMBER OF PHOTONS TO TRACE THROUGH                        LIT00060
C     NTHETA = THE NUMBER OF THETA "BINS" TO KEEP TRACK CF              LIT00070
C     NFLDVW = THE NUMBER OF "FIELD OF VIEW BINS" FOR EACH THETA BIN    LIT00080
C     NSHLS = THE NUMBER OF SHELLS TO BE INTERSECTED BY EACH PHOTON     LIT00090
C     DISTSH = THE DISTANCE BETWEEN EACH SHELL (ANC FROM ORIGIN FOR     LIT00100
C              FIRST)                                                   LITC011C
C     THKNES = PHYSICAL THICKNESS OF CLCUD IN KILOMETERS
C     G(1),G(2) = G VALUES FOR HENYEY-GREENSTEIN FUNCTION INSIDE ANC
C                 OUTSIDE OF CLOUD, RESPECTIVELY
C     RPT(1), RPT(2) = RATIOS OF PARTICULATE TO TOTAL SCATTER INSIDE AND
C                      CUTSIDE, RESPECTIVELY
C     KSCA(1), KSCA(2) = SCATTERING CCEFFICIENTS INSIDE ANC OUTSIDE OF
C                        CLOUD, RESPECTIVELY ( INVERSE KILOMETERS )
C     KABS(1), KABS(2) = ABSORPTICN COEFFICIENTS INSIDE AND OUTSIDE
C                        OF CLOUC, RESPECTIVELY ( INVERSE KILOMETERS )
C     N = NUMBER OF INPUT DATA PAIRS USED TO DEFINE AN ARBITRARY NCRM-
C         ALIZED PHASE FUNCTION
C     FTIM(I) = TIME BINS FOR TABULATING PHOTON ARRIVAL AT EDGE OF CLOUD
C     SIGTIM = VARIABLE USEC TO CALCULATE DEVIATICN OF PHOTON ARRIVAL
C              TIMES ABOUT THE MEAN FOR PLANE PARALLEL CLOUCS
C     AVGTIM = VARIABLE USED TO CALCULATE MEAN OF PHOTON ARRIVAL TIMES
C              FOR PLANE PARALLEL CLOUCS
C     PLOT = INPUT VALUE TO INDICATE IF PLOT OF CCNTOURS OF EQUAL
```

```
C        PHOTON FLUX IS DESIRED,  PLOT UNEQUAL  0  CALLS FOR PLOT
C     INS, IBANG, IFAR, IFARN = VARIABLES USED IN BOUNDARY CROSSING
C                               LOGIC
C     DCT1, DOTO = PROJECTIONS OF PHOTON POSITION ALONG DIRECTION
C                  OF INCIDENCE AT EACH COLLISION, NEW AND OLD RES-
C                  PECTIVELY.  USED IN BOUNDARY CROSSING LOGIC.
C     FBACK = RATIO OF BACKSCATTER TO TOTAL PARTICULATE SCATTERING       LIT00150
C     IX IS A RANDOM NUMBER TO START THINGS                               LIT00160
C                                                                         LIT00170
C     X(I) = X COORDINATE; I=1 FOR ORIGIN, I=2 FOR X=1, I=3 FOR Z=1       LIT00190
C          (X=1 MEANS (1,0,0) POINT FOR  ORIGINAL COORD SYST, ETC.)       LIT00200
C     Y(I) = Y COORDINATE;  DITTO                                         LIT00210
C     Z(I) = Z   "           "                                            LIT00220
C     XT(I) = THE NEW COORDINATE AFTER ROTATION BY THETA, PHI, AND        LIT00230
C          TRANSLATION BY DISTANCE DR                                     LIT00240
C     YT(I), AND ZT(I), LIKEWISE                                          LIT00250
C     ZH(I) = TEMPORARY COORDINATES HELD FOR BOUNDARY CROSSING ADJUSTMENTLIT00255
C     XS(I) IS A TEMPORARY COORDINATE POINT, AT INTERSECTION WITH SPHERELIT00260
C     YS(I), ZS(I), LIKEWISE                                              LIT00270
C     TH = VALUE OF THETA BY WHICH COORDINATE SYSTEM IS ROTATED AFTER     LIT00280
C          SIMULATED COLLISION WITH PARTICLE                              LIT00290
C     PH = VALUE OF PHI, LIKEWISE                                         LIT00300
C     THP = VALUE OF THETA PRIME, THETA RELATIVE TO PHOTON-FIXED COORD    LIT00310
C     PHP = VALUE OF PHI PRIME, IN PHOTON FIXED COORD SYST                LIT00320
C     BINS(I,J,K) = NO OF OCCURRENCES AT ITH SPHERE WITH JTH              LIT00330
C          BIN OF THETA AND KTH BIN OF THETA PRIME                        LIT00340
C     RSHL(I) = DISTANCE OF ITH SHELL FROM ORIGIN                         LIT00350
C     DTHETA = NEW VALUE OF THETA RELATIVE TO OLD DIRECTION               LIT00360
C     DPHI = NEW VALUE OF PHI RELATIVE TO OLD DIRECTION                   LIT00370
C     DR = DISTANCE OF PHOTON TRAVEL AFTER A COLLISION                    LIT00380
C     IX, IY = FIXED-POINT RANDOM NUMBERS                                 LIT00390
C     RTOT = TOTAL DISTANCE A PHOTON TRAVELS                              LIT00400
C     DIST = DISTANCE FROM ORIGIN OF PHOTON                               LIT00410
C     SDIST = DISTANCE FROM ORIGIN FOR PRIVIOUS CALCULATION               LIT00420
C     PDIST = DISTANCE ACTUALLY TRAVELED FROM ORIGIN TO SHELL             LIT00430
C     ISAV = INTEGER CORRESPONDING TO NUMBER OF SHELL BEYOND WHICH THE    LIT00440
C          PHOTON IS LOCATED                                              LIT00450
C     XFDR = RATIO OF DISTANCE FROM POINT OF COLLISION TO A SHELL         LIT00460
C          RELATIVE TO TOTAL DISTANCE A PHOTON TRAVELS                    LIT00470
C          AFTER A COLLISION                                              LIT00480
C                                                                         LIT00520
      SUBROUTINE LITE (NPHCT,NTHETA,NFLCVW,NSHLS,DISTSH,THKNES,SCA,ABB,GLIT00530
     1,RPT,FBACK,IX,BINS,BINDST,METHOD,IPRT,N,PTIM,SIGTIM,AVGTIM)         LIT00540
      INTEGER*4 BINS(10,20,1),BINDST(10,20,20)                            LIT00550
      INTEGER PLOT                                                        LIT00560
      DIMENSION G(2), SCA(2) , ABB(2), N(2), RPT(2)                       LIT00570
      DIMENSION X(4), Y(4), Z(4), XT(4), YT(4), ZT(4), XS(4), YS(4), ZS(LIT00580
     14), RSHL(20), VX(3), VY(3), VZ(4), ZH(4)                            LIT00590
```

```
      DIMENSION PTIM(80)
      DATA PI/3.1415926536/,TPI/6.2831853072/,ROOTPI/1.772453851/
      DATA EPSIL/1.00E-03/
      WRITE (6,960)
      WRITE (6,1000) NPHCT,NTHETA,NFLDVW,NSHLS,N(1),N(2),IPRT,METHOD,IX
      WRITE(6,980)DISTSH,THKNES,SCA,ABB,G,RPT,FBACK,GIN
C**** INITIALIZE ARRAYS
C
      DO 60 I=1,NSHLS
C
      DC 60 J=1,NTHETA
C
      DC 20 K=1,20
   20 BINCST(I,J,K) = 0
C
C
      DC 40 K=1,NFLDVW
   40 BINS(I,J,K) = 0
C
   6C CONTINUE
C
C
      DO 80 I=1,8C
   80 PTIM(I) = 0.0
C
      TIMT = 0.0
      TIMD = C.0
      CT = 0.0
      D = -DISTSH
      NUP = NSHLS+1
C
      DC 100 I=1,NUP
      D = D+DISTSH
  100 RSHL(I) = D
C
C**** START LOOP FOR EACH PHOTON
C
      DC 940 NPH=1,NPHOT
      IXS = IX
      NSCA = 0
      RTOT = 0.0
      ISAV = 1
C
      DO 120 I=1,4
      X(I) = 0.0
      Y(I) = 0.0
  12C Z(I) = 0.0
C
```

```
      X(2) = 1.0                                                          LIT01090
      Y(4) = 1.0                                                          LIT01100
      Z(3) = 1.0                                                          LIT01110
C*****CHOOSE DISTANCE OF INITIAL PHOTON                                   LIT01120
      INS = 1                                                             LIT01130
      IBANG = 0                                                           LIT01140
  140 TAU = 1.0/(ABB(INS)+SCA(INS))                                       LIT01150
      DR = RANEXP(IX,IY,TAU)                                              LIT01160
      IF ((DR.GT.THKNES).AND.(INS.EQ.1)) GO TO 200                        LIT01170
C                                                                         LIT01180
      DO 160 I=1,4                                                        LIT01190
  160 Z(I) = Z(I)+DR                                                      LIT01200
C                                                                         LIT01210
  180 IF (Z(1).LT.RSHL(ISAV+1)) GO TO 240                                 LIT01220
      ISAV = ISAV+1                                                       LIT01230
      IF (ISAV.GT.NSHLS) GO TO 940                                        LIT01240
      GO TO 180                                                           LIT01250
  200 DR = THKNES                                                         LIT01260
C                                                                         LIT01270
      DO 220 I=1,4                                                        LIT01280
  220 Z(I) = Z(I)+DR                                                      LIT01290
C                                                                         LIT01300
      INS = 2                                                             LIT01310
      GO TO 140                                                           LIT01320
  240 CONTINUE                                                            LIT01330
      RTOT = Z(1)                                                         LIT01340
      SDIST = RTOT                                                        LIT01350
      T = THKNES                                                          LIT01360
C*****DETERMINE IF PHOTON IS ABSORBED OR SCATTERED                         LIT01370
  260 RATIO = SCA(INS)/(ABB(INS)+SCA(INS))                                LIT01380
      CALL RANDU (IX,IY,RN)                                               LIT01390
      IX = IY                                                             LIT01400
      IF (RN.GT.RATIO) GO TO 940                                          LIT01410
C**** IF PHOTON IS SCATTERED, DETERMINE NEW ANGLES AND DISTANCES           LIT01420
      NSCA = NSCA+1                                                       LIT01430
      DR = RANEXP(IX,IY,TAU)                                              LIT01440
      DTHETA = RANTH(IX,IY,G,RPT,FBACK,N,INS)                             LIT01450
      DPHI = RANPH(IX,IY)                                                 LIT01460
      STH = SIN(DTHETA)                                                   LIT01470
      CTH = COS(DTHETA)                                                   LIT01480
      SPH = SIN(DPHI)                                                     LIT01490
      CPH = COS(DPHI)                                                     LIT01500
C**** ROTATE TO NEW PHOTON POSITION AND TRANSLATE                          LIT01510
C                                                                         LIT01520
      DO 280 I=1,4                                                        LIT01530
      XT(I) = X(I)*CTH*CPH-Y(I)*CTH*SPH-Z(I)*STH                          LIT01540
      YT(I) = X(I)*SPH+Y(I)*CPH                                           LIT01550
C**** HOLD PRESENT Z COORDINATES UNTIL BOUNDARY CROSSING DETERMINATION      LIT01555
```

```
C       IS MADE                                                          LIT01556
        ZH(I) = X(I)*STH*CPH-Y(I)*STH*SPH+Z(I)*CTH                       LIT01560
        ZT(I) = ZH(I)+DR                                                 LIT01570
  280 CONTINUE                                                           LIT01580
C                                                                        LIT01590
C**** DETERMINE IF A BOUNDARY HAS BEEN CROSSED                            LIT01595
        DISTO = SQRT(X(1)**2+Y(1)**2+Z(1)**2)                            LIT01600
        VALO = (X(1)*(X(1)-X(3))+Y(1)*(Y(1)-Y(3))+Z(1)*(Z(1)-Z(3)))/DISTO LIT01610
        IF (VALO.GT.1.0) VALO=1.0                                        LIT01620
        IF (VALO.LT.-1.0) VALO=-1.0                                      LIT01630
        THEO = PI-ARCOS(VALO)                                            LIT01640
        IF (THEO.LT.0.0) THEO=0.0                                        LIT01650
  300 DIST1 = SQRT(XT(1)**2+YT(1)**2+ZT(1)**2)                           LIT01660
        VALN = (XT(1)*(XT(1)-XT(3))+YT(1)*(YT(1)-YT(3))+ZT(1)*(ZT(1)-ZT(3)LIT01670
      1))/DIST1                                                          LIT01680
        IF (VALN.GT.1.0) VALN=1.0                                        LIT01690
        IF (VALN.LT.-1.0) VALN=-1.0                                      LIT01700
        THEN = PI-ARCOS(VALN)                                            LIT01710
        IF (THEN.LT.0.0) THEN=0.0                                        LIT01720
C******CHECK TO SEE IF PHOTON HAS CROSSED A BORDER                        LIT01730
C                                                                        LIT01740
        DOTO = DISTO*COS(THEO)                                           LIT01750
C                                                                        LIT01760
        DOT1 = DIST1*COS(THEN)                                           LIT01770
        IF (DOTO.GT.THKNES) IFAR=3                                       LIT01780
        IF ((DOTO.LE.THKNES).AND.(THEO.LT.PI/2.0)) IFAR=2                LIT01790
        IF (THEO.GE.PI/2.0) IFAR=1                                       LIT01800
        IF (DOT1.GT.THKNES) IFARN=3                                      LIT01810
        IF ((DOT1.LE.THKNES).AND.(THEN.LT.PI/2.0)) IFARN=2               LIT01820
        IF (THEN.GE.PI/2.0) IFARN=1                                      LIT01830
        IF (IBANG.EQ.1) GO TO 380                                        LIT01840
        IF (IBANG.EQ.2) GO TO 500                                        LIT01850
        IF (IFARN.NE.2) GO TO 560                                        LIT01860
C*****NEW PHOTON POSITION IS INSIDE SCATTERING CLOUD                      LIT01870
        IF (IFAR.EQ.2) GO TO 700                                         LIT01880
C*****PHOTON PASSED FROM OUTSIDE TO INSIDE OF SCATTERING CLOUD            LIT01890
        INS = 1                                                          LIT01900
        IF (IFAR.EQ.1) GO TO 440                                         LIT01910
C******PHOTON PASSED FROM OUTSIDE THICK TO INSIDE THICK                   LIT01920
  320 FRACT = (T-DOT1)/(DOTO-DOT1)                                       LIT01930
C                                                                        LIT01940
        DO 340 J=1,4                                                     LIT01950
  340 ZT(J) = ZT(J)-FRACT*DR                                             LIT01960
C                                                                        LIT01970
        TAU = 1.0/(ABB(INS)+SCA(INS))                                    LIT01980
        DR = RANEXP(IX,IY,TAU)                                           LIT01990
C                                                                        LIT02000
        DO 360 J=1,4                                                     LIT02010
```

```
  360 ZT(J) = ZT(J)+DR
C
      IBANG = 1
      GO TO 300
  380 IBANG = 0
      IF (IFARN.EQ.2) GO TO 700
C*****PHOTON PASSED FROM OUTSIDE TO INSIDE ,WAS CHECKED THEN PASSED
C     OUTSIDE AGAIN SO IT NEEDS TO BE CHECKED AGAIN
      INS = 2
      FRACT = -DOT1/(T-DOT1)
C
      DO 400 I=1,4
  400 ZT(I) = ZT(I)-FRACT*DR
C
      TAU = 1.0/(ABB(INS)+SCA(INS))
      DR = RANEXP(IX,IY,TAU)
C
      DO 420 I=1,4
  420 ZT(I) = ZT(I)+DR
C
      GO TO 700
C*****PHOTON PASSED FROM BEHIND CLOUD TO WITHIN CLOUD
C     OR PHOTON PASSED FROM BEHIND CLOUD COMPLETELY THROUGH CLOUD
  440 FRACT = DOT1/(DOT1-DOT0)
C
      DO 460 I=1,4
  460 ZT(I) = ZT(I)-FRACT*DR
C
      TAU = 1.0/(ABB(INS)+SCA(INS))
      DR = RANEXP(IX,IY,TAU)
C
      DO 480 J=1,4
  480 ZT(J) = ZT(J)+DR
C
      IBANG = 2
      GO TO 300
  500 IBANG = 0
      IF (IFARN.EQ.2) GO TO 700
C*****PHOTON PASSED FROM OUTSIDE TO INSIDE THEN CHECKED THEN IT PASSED
C*****OUTSIDE AGAIN AND A SECOND CHECK IS NEEDED
      INS = 2
      FRACT = (DOT1-T)/DOT1
C
      DO 520 I=1,4
  520 ZT(I) = ZT(I)-FRACT*DR
C
      CT = CT+1.0
      TIM = ((RTOT+(1-FRACT)*DR)-T)/3.0E+05
```

```
      IF (TIM.LT.1.0E-09) TIM=1.0E-09
      IF (TIM.GT.1.0E-03) TIM=.99E-03
      NBIN = (ALOG10(TIM)+9.0)*13.3333+1
      IF (NBIN.LT.1) NBIN=1
      PTIM(NBIN) = PTIM(NBIN)+1.0
      TIMT = TIMT+TIM
      TIMD = TIMD+TIM**2
      TAU = 1.0/(ABB(INS)+SCA(INS))
      DR = RANEXP(IX,IY,TAU)
C
      DO 540 I=1,4
  540 ZT(I) = ZT(I)+DR
C
      GO TO 700
C*****NEW PHOTON POSITION IS OUTSIDE OF SCATTERING CLOUD
  560 IF ((IFAR.EQ.1).OR.(IFAR.EQ.3)) GO TO 680
C*****PHOTON PASSED FROM INSIDE CLOUD TO OUTSIDE CLOUD
      INS = 2
      IF (IFARN.EQ.1) GO TO 620
C*****PHOTON PASSED FROM WITHIN CLOUD TO A POSITION GREATER THAN THKNES
      FRACT = (DOT1-T)/(DOT1-DOT0)
C
      DO 580 I=1,4
  580 ZT(I) = ZT(I)-FRACT*DR
C
      CT = CT+1.0
      TIM = ((RTOT+(1-FRACT)*DR)-T)/3.0E+05
      IF (TIM.LT.1.0E-09) TIM=1.0E-09
      IF (TIM.GT.1.0E-03) TIM=.99E-03
      NBIN = (ALOG10(TIM)+9.0)*13.3333+1
      IF (NBIN.LT.1) NBIN=1
      PTIM(NBIN) = PTIM(NBIN)+1.0
      TIMT = TIMT+TIM
      TIMD = TIMD+TIM**2
      TAU = 1.0/(ABB(INS)+SCA(INS))
      DR = RANEXP(IX,IY,TAU)
C
      DO 600 I=1,4
  600 ZT(I) = ZT(I)+DR
C
      GO TO 700
C*****PHOTON PASSED FROM WITHIN CLOUD TO A POSITION BEHIND CLOUD
  620 FRACT = -DOT1/(DOT0-DOT1)
C
      DO 640 I=1,4
  640 ZT(I) = ZT(I)-FRACT*DR
C
      TAU = 1.0/(ABB(INS)+SCA(INS))
```

```
      DR = RANEXP(IX,IY,TAU)
C
      DO 660 I=1,4
  660 ZT(I) = ZT(I)+DR
C
      GO TO 700
C*****PHOTON PASSED FROM OUTSIDE CLOUD TC OUTSIDE CLOUD
  680 IF (IFAR.EQ.IFARN) GO TO 700
C*****PHOTON PASSEC COMPLETELY THROUGH CLOUD
      IF (IFARN.EQ.3) GO TO 440
C*****PHOTON PASSEC FROM FRCNT OF CLOUD TO BEHINC CLCUD
      GC TO 320
C*****BORDER WAS NOT CROSSED OR CHECKING HAS BEEN ACCOMPLISHED
  7CC DR = SQRT((ZH(1)-ZT(1))**2)
      CIST = SQRT(XT(1)**2+YT(1)**2+ZT(1)**2)
      FF = 1.0-ZT(1)/DR
C**** CHECK TO SEE IF PHCTON HAS PENETRATED A SHELL
C**** CCNSIDER IF PHOTON HAS PENETRATED IN AND OUT OF SHELL
C     FF IS FRACTIONAL DISTANCE BETWEEN TWO POINTS GIVING
C         THE DISTANCE OF CLCSEST APPROACH
      IF ((FF.LT.0.0).OR.(FF.GT.1.0)) GO TO 720
C     DC IS DISTANCE CF CLOSEST APPROACH
      DC = SQRT(XT(1)**2+YT(1)**2)
      IF (DC.GT.RSHL(ISAV)) GO TO 720
      ISAV = ISAV-1
      IF (DC.GT.RSHL(ISAV)) GO TO 800
      GC TO 720
C**** CCNSIDER IF PHOTON HAS PENETRATED CUTER SHELLS
  720 IF ((ISAV).GT.(NSHLS)) GO TO 940
      IF (DIST.GT.RSHL(ISAV+1)) GC TO 8CC
  740 IF (DIST.GT.RSHL(ISAV)) GO TO 760
      ISAV = ISAV-1
      GO TO 740
C**** TIME TO UPDATE COORDINATES - GET READY FOR ANOTHER CCLLISICN
  760 RTOT = RTOT+DR
      SDIST = DIST
C
      DO 780 I=1,4
      X(I) = XT(I)
      Y(I) = YT(I)
      Z(I) = ZT(I)
  780 CCNTINUE
C
C**** GC TO 20 FOR ANOTHER COLLISION
      GC TO 260
C**** CALCULATE COORDINATES OF POINT AT WHICH PHOTON PENETRATES SHELL
  800 ISAV = ISAV+1
      XFDR = (DR-ZT(1)+SQRT((ZT(1)-DR)**2-SDIST**2+RSHL(ISAV)**2))/DR
```

```
C
      DO 820 I=1,4
      XS(I) = XT(I)
      YS(I) = YT(I)
      ZS(I) = ZT(I)-(1.0-XPDR)*DR
  820 CONTINUE
C
      PDIST = RTOT+XPDR*DR
C**** CALCULATE VALUE OF THETA
      VAL = ((XS(1)*(XS(1)-XS(3))+YS(1)*(YS(1)-YS(3))+ZS(1)*(ZS(1)-ZS(3)
     1))/RSHL(ISAV))
      IF(ABS(VAL).GT.1.1) CALL PERR(1,IXS,VAL,XS,YS,ZS)
      IF (ABS(VAL).GT.1.1) WRITE (6,1020) VAL,NPH,IXS
      IF (VAL.GT.1.0) VAL=1.0
      IF (VAL.LT.-1.0) VAL=-1.0
      TH = PI-ARCOS(VAL)
      IF (TH.LT.0.0) TH = 0.0
C**** CALCULATE VALUE OF PHI
      DENOMS = (RSHL(ISAV)**2-(XS(1)*(XS(3)-XS(1))+YS(1)*(YS(3)-YS(1))+Z
     1S(1)*(ZS(3)-ZS(1)))**2)
      IF(DENOMS.LT.-EPSIL) CALL PERR(5,IXS,DENOMS,XS,YS,ZS)
      IF (DENOMS.LT.EPSIL) DENOMS=0.0
      DENOM = SQRT(DENOMS)
      IF (DENOM.EQ.0.0) PH=0.0
      IF (DENOM.EQ.0.0) GO TO 840
      VAL = ((XS(1)*(XS(1)-XS(2))+YS(1)*(YS(1)-YS(2))+ZS(1)*(ZS(1)-ZS(2)
     1))/DENOM)
      IF(ABS(VAL).GT.1.1) CALL PERR(2,IXS,VAL,XS,YS,ZS)
      IF (ABS(VAL).GT.1.1) WRITE (6,1020) VAL,NPH,IXS
      IF (VAL.GT.1.0) VAL=1.0
      IF (VAL.LT.-1.0) VAL=-1.0
      PH = ARCOS(VAL)
      IF ((XS(1)*(XS(1)-XS(4))+YS(1)*(YS(1)-YS(4))+ZS(1)*(ZS(1)-ZS(4))).
     1LT.0.0) PH=TPI-PH
  840 CONTINUE
C**** CALCULATE VALUE OF THETA PRIME (RECEIVER)
      VAL = (ZS(1)/RSHL(ISAV))
      IF(ABS(VAL).GT.1.1) CALL PERR(3,IXS,VAL,XS,YS,ZS)
      IF (ABS(VAL).GT.1.1) WRITE (6,1020) VAL,NPH,IXS
      IF (VAL.GT.1.0) VAL=1.0
      IF (VAL.LT.-1.0) VAL=-1.0
      THP = ARCOS(VAL)
C**** CALCULATE VALUE OF PHI PRIME (RECEIVER)
      PARAM = (XS(1)*XS(3)+YS(1)*YS(3)+ZS(1)*ZS(3))/(RSHL(ISAV)*SQRT(XS(
     13)**2+YS(3)**2+ZS(3)**2))
      IF (PARAM.LT.0.999) GO TO 860
      VAL = XS(2)-XS(1)
      GO TO 1040
```

```
  860 BP = -RSHL(ISAV)**2/(XS(3)*XS(1)+YS(3)*YS(1)+ZS(3)*ZS(1))
      VX(1) = XS(1)+BP*XS(3)
      VY(1) = YS(1)+BP*YS(3)
      VZ(1) = ZS(1)+BP*ZS(3)
      ANORM = SQRT(VX(1)**2+VY(1)**2+VZ(1)**2)
      VX(1) = VX(1)/ANORM
      VY(1) = VY(1)/ANORM
      VZ(1) = VZ(1)/ANORM
      VX(3) = XS(1)/RSHL(ISAV)
      VY(3) = YS(1)/RSHL(ISAV)
      VZ(3) = ZS(1)/RSHL(ISAV)
      VX(2) = VY(3)*VZ(1)-VZ(3)*VY(1)
      VY(2) = VZ(3)*VX(1)-VX(3)*VZ(1)
      VZ(2) = VX(3)*VY(1)- VY(3)*VX(1)
      IF (VZ(3).LT.0.9999) GO TO 880
      VAL = 1.0
      GO TO 900
  880 CONTINUE
      VAL = VZ(1)/SQRT(1.0-VZ(3)**2)
 1040 IF(ABS(VAL).GT.1.1) CALL PERR(4,3,RSHL(ISAV),VX,VY,VZ)
      IF(ABS(VAL).GT.1.1) CALL PERR(4,IXS,VAL,XS,YS,ZS)
      IF (ABS(VAL).GT.1.1) WRITE (6,1020) VAL,NPH,IXS
      IF (VAL.GT.1.0) VAL=1.0
      IF (VAL.LT.-1.0) VAL=-1.0
  900 PHP = ARCOS(VAL)
      IF (VZ(2).LT.0.0) PHP=TPI-PHP
C**** TALLY LOCATION OF ANGULAR RESULTS
      IF (METHOD.LE.2) NTH=(TH*NTHETA/PI)+1
      IF (METHOD.GE.3) NTH=(SQRT(TH)*NTHETA)/ROOTPI+1
      IF (METHOD.LE.2) NTHP=2.0*THP*NFLDVW/PI+1
      IF (METHOD.GE.3) NTHP=(SQRT(2.0*THP)*NFLDVW)/ROOTPI+1
      IF (IPRT.NE.9) GO TO 920
  920 CONTINUE
      BINS(ISAV-1,NTH,NTHP)=BINS(ISAV-1,NTH,NTHP)+1
CX    XXXX = ((YS(3)-YS(1))*ZS(1) - (ZS(3)-ZS(1))*YS(1))*VX(1)
CX    YYXX = ((ZS(3)-ZS(1))*XS(1) - (XS(3)-XS(1))*ZS(1))*VY(1)
CX    ZZXX = ((XS(3)-XS(1))*YS(1) - (YS(3)-YS(1))*XS(1))*VZ(1)
C**** TALLY LOCATION OF INTERSECTION IN DISTANCE OF ARRIVAL BINSA
      IF (PDIST.LE.(1.00001*RSHL(ISAV))) PDIST=1.00001*RSHL(ISAV)
      XDIST = ALOG10((PDIST/RSHL(ISAV))-1.0)+4.0
      IF (XDIST.LT.0.0) XDIST=0.0
      IF (XDIST.GE.4.99) XDIST=4.99
      NDIST = (XDIST*20)/5+1
      BINDST(ISAV-1,NTH,NDIST)=BINDST(ISAV-1,NTH,NDIST)+1
      GO TO 720
  940 CONTINUE
C
      AVGTIM = TIMT/CT
```

```
      SIGTIM = SQRT(TIMD/CT-AVGTIM**2)
      RETURN
C
  96C FCRMAT (' ENTERED LITE')
  980 FORMAT (' DISTSH= ',F6.3,5X,'THICKNESS= ',F6.3,3X,'SCA(1)= ',F6.3,
     1 6X,'SCA(2)= ',F6.3,/,' ABB(1)= ',F6.3,5X,'ABB(2)= ',F6.3,6X,
     2 'G(1)= ',F6.3,8X,'G(2)= ',F6.3,/,' RPT(1)= ',F6.3,5X,'RPT(2)= ',
     3F6.3,5X,'FBACK= ',F6.3,6X,'GIN= ',D6.3,/)
 1000 FORMAT (/,' NPHOT=',I5,9X,' NTHETA=',I5,3X,'NFLDVW=',I5,8X,'NSHS=',
     1 I5,/,' N(1)=',I5,10X,'N(2)=',I5,10X,'IPRT=',I5,10X,'METHCD=',I5,/
     2 ' IX= ',I10,/)
 1020 FORMAT (' ARCOS GT. 1.1; VAL = ',F10.5,'  NPHCT = ',I5,'  IX = ',
     1  I10)
      ENC
C
      SUBRCUTINE PERR  (ERRNO,IXS,PARAM,XS,YS,ZS)
      INTEGER *4ERRNO
      DIMENSION XS(4), YS(4), ZS(4)
      NUP = 4
      IF (IXS.LT.4) NUP = IXS
      WRITE (6,20) ERRNC,PARAM,IXS
      WRITE (6,40) (XS(I),YS(I),ZS(I),I=1,NUP)
      RETURN
C
   20 FCRMAT (' ERROR DETECTED, LOCATION NO. ',I5,' PARAM = ',
     1  E14.8,' IXS = ',I10)
   40 FCRMAT (' COORD. FROM PERR: ',12F9.5)
      END


C**** FUNCTION SUBROUTINE RANEXP ****
C
      FUNCTION RANEXP (IX,IY,TAU)
C**** THIS FUNCTION GENERATES A RANDCM NUMBER WEIGHTED EXPONENTIALLY
      CALL RANDU (IX,IY,RN)
      IX = IY
      RANEXP = -TAU*ALOG(1.0-RN)
      RETURN
      END
```

```
C**** FUNCTION SUBROUTINE RANTH
C
C**** THIS FUNCTION GENERATES A RANDOM VALUE OF THETA,
C        SUITABLY WEIGHTED
      FUNCTION RANTH (IX,IY,G,RPT,FBACK,N,INS)
      DIMENSION N(2), G(2), THETA(2,70), PHASE(2,70), SLOPE(2,70), WEIGH
     1T(2,70), RPT(2)
      COMMON WEIGHT,PHASE,SLOPE,THETA
      DATA PI2/1.5707963268/,PI/3.1415926536/
      CALL RANDU (IX,IY,RN)
      IX = IY
      IF (RN.GT.RPT(INS)) GO TO 80
      IF (FBACK.EQ.0.0) GO TO 60
      CALL RANDU (IX,IY,RN)
      IX = IY
      CRANM1 = 0.0
C
      DO 20 I=1,8
      CRANTH = HENA-HENB/(RN-HENC-HEND*FBACK*CRANM1*(1.0-CRANM1**2))**2
      IF (ABS(CRANTH-CRANM1).LT.0.001) GO TO 40
      CRANM2 = CRANM1
      CRANM1 = CRANTH
   20 CONTINUE
C
      CRANTH = 0.25*(CRANM2+CRANTH)+0.5*CRANM1
   40 CONTINUE
      IF (ABS(CRANTH).GT.1.0) CRANTH=CRANTH/(ABS(CRANTH)+.00001)
      RANTH = ARCOS(CRANTH)
      RETURN
C**** CALCULATE THETA USING FORWARD SCATTER FUNCTION HERE
   60 IF (N(1).NE.0) GO TO 100
      CALL RANDU (IX,IY,RN)
      IX = IY
      HENA = (1.0+G(INS)**2)/(2.0*G(INS))
      HENB = (1.0-G(INS)**2)**2/(8.0*G(INS)**3)
      HENC = (G(INS)+1.0)/(2.0*G(INS))
      HEND = (1.0-G(INS)**2)/(4.0*(1.0+G(INS)**2)**1.5)
      RANTH = ARCOS(HENA-HENB/(HENC-RN)**2)
      RETURN
C**** CALCULATE THETA ASSUMING RAYLEIGH SCATTERING HERE
   80 CALL RANDU (IX,IY,RN)
      IX = IY
      RLB = 4.0*RN-2.0
      RLBS = SQRT(RLB**2+1.0)
      CAPA = (-RLB+RLBS)**(1./3.)
      CAPB = -(RLB+RLBS)**(1./3.)
      RANTH = ARCOS(CAPA+CAPB)
      RETURN
C**** CALCULATE THETA BY USE OF INPUTTED DATA PAIRS HERE
```

```
  100 CALL RANDU (IX,IY,RN)
      IX = IY
      W1 = 0.0
      W2 = WEIGHT(INS,1)
      M = N(INS)-1
C
      DO 120 I=1,M
      NIT = I
      IF ((RN.GE.W1).AND.(RN.LT.W2)) GO TO 140
      W1 = W2
      W2 = W2+WEIGHT(INS,I+1)
  120 CONTINUE
C
  140 RN = (RN-W1)/(W2-W1)
      P = PHASE(INS,NIT)
      B = SLOPE(INS,NIT)
      T = THETA(INS,NIT)*PI/180.0
      T1 = THETA(INS,NIT+1)*PI/180.0
      A = P-B*T
      C = RN*(A*T1+B*T1**2/2.0)+(1.0-RN)*(B*T*#2/2.0+A*T)
      IF (B.GE.0.0) GO TO 160
      RANTH = -A/B-SQRT((A/B)**2+2.0*C/B)
      RETURN
C
  160 RANTH = -A/B+SQRT((A/B)**2+2.0*C/B)
      RETURN
      END



C**** FUNCTION SUBROUTINE RANPH ****
C
C**** THIS FUNCTION GENERATES A RANDOF VALUE OF PHI
      FUNCTION RANPH (IX,IY)
      DATA TPI/6.283183072/
      CALL RANDU (IX,IY,RN)
      IX = IY
      RANPH = TPI*RN
      RETURN
      END
```

```
C**** EFFECTIVE ATTENUATION COEFFICIENT PROGRAM
C
C      THIS PROGRAM CALCULATES THE EFFECTIVE ATTENUATION COEFFICIENT, THE
C      RELATIVE FLUX THROUGH AN APERTURE OF GIVEN HALF ANGLE AND RANGE
C      AND THE BEAM SPREAD FUNCTION USING MULTIPLE GAUSSIAN QUADRATURE.
C      INPUTS REQUIRED ARE AS FOLLOWS:
C         G = HENYEY-GREENSTEIN PARAMETER OF PHASE FUNCTION USED
C         TH = INITIAL VALUE OF HALF ANGLE OF APERTURE
C         THI = INCREMENT IN TH
C         NTHN = NUMBER OF TH VALUES TO BE USED, INTEGER
C         R =  INITIAL RANGE IN UNITS OF EXTINCTION LENGTHS
C         RI = INCREMENT IN R
C         NRN = NUMBER OF R VALUES TO BE USED, INTEGER
C         NACC = ACCURACY OF NUMERICAL INTEGRATION-DOUBLE PRECISION
C           1 = 32 PT GAUSSIAN QUADRATURE
C           2 = 64 PT     "        "
C           ................
C          10 = 320      ETC.
C         SCA = SCATTERING COEFFICIENT IN INVERSE KILOMETERS
C         EXT = EXTINCTION COEFFICIENT IN INVERSE KILOMETERS
C
      DIMENSION HBSF(20,20), ALPHA(20), THETA(20), RANM(20,20), RHBSF(20
     1,20), C(20), FLUX(20,20), X(500), Y(500), Z(500)
      DOUBLE PRECISION HBSF,ALPHA,THETA,RANM,RHBSF,RANGE,THTA,G,SCA,EXT,
     1ACC,FL,XU,XL,YR,HINT
      COMMON THTA,G
      REAL *8TL(12)/'PLOTOF','NEGBSF','MAMI','LLBACH',8*''/
      INTEGER NA(10)/1,0,0,0,1,1,0,2,0,2/
      REAL*8 XAX/'KILOMTRS'/
      REAL*8 YAX/'KILOMTRS'/
      INTEGER *4IB(144),NB(1)
      EXTERNAL FCC.FCT
      DATA PI/3.1415926536D0/
   20 WRITE (6,300)
      READ (5,320) SCA,EXT,G
      WRITE (6,320) SCA,EXT,G
      IF (SCA.EQ.0.0) GO TO 280
      WRITE (6,340)
      READ (5,360) TH,THI,NTHN,R,RI,NRN,NACC,PLCT
      WRITE (6,360) TH,THI,NTHN,R,RI,NRN,NACC,PLOT
      ACC = NACC
C
      DO 100 I=1,NTHN
      FL = 0.0
      THTA = (TH+(I-1)*THI)*PI/180.0
C
      DO 40 J=1,NACC
      XU = J/ACC
      XL = (J-1)/ACC
```

```
      CALL DQG32  (XL,XU,FCT,YR)
      FL = FL+YR
   40 CONTINUE
C
      ALPHA(I) =  EXT-SCA*FL
      HINT = 0.0
C
      DO 60 L=1,NACC
      XL = DATAN(THTA)+((L-1)/ACC)*(PI/2.0-DATAN(THTA))
      XU = DATAN(THTA)+(L/ACC)*(PI/2.0-DATAN(THTA))
      CALL DQG32  (XL,XU,FCC,YR)
      HINT = HINT +YR
   60 CONTINUE
C
C
      DO 80 K=1,NRN
      RAN =  R+(K-1)*RI
      FLUX(I,K)  =  DEXP(-ALPHA(I)*RAN/EXT)
      HBSF(I,K)  =  SCA*FLUX(I,K)*HINT/(RAN*DTAN(THTA)*EXT)
      RHBSF(I,K)  = -DLOG10(HBSF(I,K))
      RANM(I,K)  =  RAN/DCOS(THTA)
   80 CONTINUE
C
  100 CONTINUE
C
      WRITE (6,380)
C
      DO 120  I=1,NTHN
      THETA(I)  =  ((I-1)*THI)+TH
      WRITE (6,400)  THETA(I), ALPHA(I)
  120 CONTINUE
C
      WRITE (6,420)
      WRITE (6,440)  ((FLUX(I,J),I=1,NTHN,2),J=1,NRN,2)
      WRITE (6,460)
      WRITE (6,440)  ((HBSF(I,J),I=1,NTHN,2),J=1,NRN,2)
      WRITE (6,480)
      WRITE (6,500)  ((RHBSF(I,J),I=1,NTHN,2),J=1,NRN,2)
C     NEXT LINES USED IF PLOT WAS REQUESTED BY SETTING PLOT >  0
      IF (PLOT.NE.0.0) GO TO 140
      STOP
C
  140 DO 160  I=1,NTHN
C
      DO 160  K=1,NRN
      Y((I-1)*NRN+K)  =  RANM(I,K)*DSIN(THETA(I)*PI/180.0)
      X((I-1)*NRN+K)  =  RANM(I,K)*DCOS(THETA(I)*PI/180.0)
      Z((I-1)*NRN+K)  =  RHBSF(I,K)
```

```
  160 CONTINUE
C
      N = NRN*NTHN
C
      DO 180 I=1,NTHN
      IB(I) = (I-1)*NRN+1
  180 CONTINUE
C
      M = NRN-1
C
      DO 200 I=1,M
      IB(NTHN+I) = (NTHN-1)*NRN+1+I
  200 CONTINUE
C
      M1 = NTHN-1
C
      DO 220 I=1,M1
      IB(NTHN+NRN-1+I) = (NTHN-I)*NRN
  220 CONTINUE
C
      M2 = NRN-2
C
      DO 240 I=1,M2
      IB(2*NTHN+NRN-2+I) = NRN-I
  240 CONTINUE
C
      NR = 1
      NB(1) = 2*NTHN+2*NRN-4
      NC = 20
C
      DO 260 I=1,NC
      C(I) = -1.0+I*.5
  260 CONTINUE
C
      CF = .5
      XS = -5.0
      YS = -5.0
      CALL CONISD (X,Y,Z,N,IB,NR,NB,C,NC,CF,XS,YS,TL,XAX,YAX,NA,IER)
      WRITE (6,520) IER
      GO TO 20
  280 CONTINUE
      STOP
C
  300 FORMAT (/,'  INPUT SCA,EXT,G, USING 3F6,3',/)
  320 FORMAT (3F6.3)
  340 FORMAT (/,' INPUT  TH,THI,NTHN, F,RI,NRN,NACC,PLOT USING 2(2F6.3,I3
     1),I3,I1',/)
  360 FORMAT (2(2F6.3,I3),I3,I1)
```

```
380 FORMAT (/,'  EFFECTIVE ATTENUATION COEFFICIENTS ARE ',/)
400 FORMAT (2F12.3)
420 FORMAT (/,' R FLUX  IS',/)
440 FORMAT (10E12.3)
460 FORMAT (/,' HBSF IS ',/)
480 FORMAT (/,' NEG LOG OF HBSF IS ',/)
500 FORMAT (10F12.3)
520 FORMAT (/,'ERROR IER RETURNED=',I5,/)
    END
    DOUBLE PRECISION FUNCTIONFCT(X)
    IMPLICIT REAL*8(A-H,O-Z)
    COMMON THTA,G
    DEN1 = 1.0+DTAN(THTA)**2/((1.0-X)**2)
    DEN2 = DSQRT(1.0+G**2-2.0*G/DEN1)
    DM1 = (1.0-G**2)/(2.0*G)
    DM2 = (1.0/(1.0-G))-1.0/DEN2
    FCT = DM1*DM2
    RETURN
    END
    DOUBLE PRECISION FUNCTIONFCC(X)
    IMPLICIT REAL*8(A-H,O-Z)
    COMMON THTA,G
    DATA PI/3.1415926536D0/
    FCC = ((1.0-G**2)/(4.0*PI*(1.0+G**2-2.0*G*DCOS(X))**1.5))*DCOS(X)
    RETURN
    END
```

LIST OF REFERENCES

1. Bucher, E.A., "Computer Simulation of Light Pulse Propagation for Communication through Thick Clouds," Applied Optics, V. 12, p. 2391-2400, October 1973.

2. Plass, G.N. and Kattawar, G.W., "Monte Carlo Calculations of Light Scattering from Clouds," Applied Optics, V. 7, p. 415-419, March 1968.

3. Plass, G.N. and Kattawar, G.W., "Radiant Intensity of Light Scattered from Clouds," Applied Optics, V. 7, p. 699-704, April 1968.

4. Plass, G.N. and Kattawar, G.W., "Influence of Particle Size Distribution on Reflected and Transmitted Light from Clouds," Applied Optics, V. 7, p. 869-878, May 1968.

5. Plass, G.N. and Kattawar, G.W., "Radiance and Polarization of Multiple Scattered Light from Haze and Clouds," Applied Optics, V. 7, p. 1519-1527, August 1968.

6. Junge, D.M., Non-Line-of-Sight Electro-Optics Laser Communications in the Middle Ultraviolet, M.S. Applied Science Thesis, Naval Postgraduate School, Monterey, California, 1977.

7. Danielson, R.E., Moore, D.R. and van de Hulst, H.C., "The Transfer of Visible Radiation through Clouds," Journal of the Atmospheric Sciences, V. 26, p. 1078-1087, September 1969.

8. Arnush, D., "Underwater Light-beam Propagation in the Small-angle-scattering Approximation," Journal of the Optical Society of America, V. 62, p. 1109, 1972.

9. Gordon, A., "Practical Approaches to Underwater Multiple-Scattering Problem," Ocean Optics, V. 64, p. 85-93, 1975.

10. Stotts, L.B., "The Radiance Produced by Laser Radiation Transversing a Particulate Multiple-scattering Medium," Journal of the Optical Society of America, V. 67, p. 815-819, 1977.

11. Hansen, J.E., "Exact and Approximate Solutions for Multiple Scattering by Cloudy and Hazy Planetary Atmospheres," Journal of the Atmospheric Sciences, V. 26, p. 478-487, September 1968.

12. Ishimaru, A. and Hong, S.T., "Multiple Scattering Effects on Coherent Bandwidth and Pulse Distortion of a Wave Propagating in a Random Distribution of Particles," Radio Science, V. 10, p. 637-644, June 1975.

13. Kennedy, R.S., "Communication through Optical Scattering Channels: An Introduction," Proceedings of IEEE, V. 58, p. 1651-1664, October 1970.

14. Lutomirski, R.F. and Yura, H.T., "Propagation of an Optical Beam in an Inhomogeneous Medium," Applied Optics, V. 10, p. 1954, July 1971.

15. Dell-Imagine, R.A., "A Study of Multiple Scattering of Optical Radiations with Applications to Laser Communications," Advances in Communication Systems, Vol. II, V. 2, p. 1, 1966.

16. Zachor, A.S., "Aureole Radiance Field About a Source in a Scattering-Absorbing Medium," submitted for publication.

17. Mooradian, G.C., Atmospheric and Space Optical Communications for Naval Applications, paper presented at Proceedings of 6th DOD Conference on Laser Technology, Washginton, D.C., March 1974.

18. van de Hulst, H.C., Light Scattering by Small Particles, p. 1, Wiley, 1953.

19. Kerker, M., The Scattering of Light and Other Electromagnetic Radiation, p. 1, Academic, 1969.

20. Chandrasekhar, S., Radiative Transfer, Dover, 1960.

21. Deirmendjian, D., Electromagnetic Scattering on Spherical Polydispersions, American Elseveir, 1969.

22. McCartney, E.J., Optics of the Atmosphere, Wiley, 1976.

23. Sekera, Z., Advances in Geophysics, Vol. 3, Academic, 1975.

24. Born, M. and Wolf, E., Principles of Optics, Pergamon, 1975.

25. Heggastad, H.M., "Multiple Scattering for Light Transmission through Optically Thick Clouds," Journal of the Optical Society of America, V. 61, p. 1293-1300, 1971.

26. Stotts, L.B., "Closed Form Expression for Optical Pulse Broadening in Multiple-Scattering Media," Applied Optics, V. 17, p. 504-505, February 1978.

27. Dolin, L.S., "Scattering of a Light Beam in a Layer of a Turbid Medium," Bull (IZV), Radiophysics, V. 7, p. 1, 1964.

28. Naval Undersea Center Technical Project 371, Underwater Multiple Scattering of Light for System Designers, Part I, by Gordon, A., November 1973.

29. Scripts Institute of Oceanography Reference 71-1, Underwater Lighting by Submerged Lasers and Incandescent Sources, by Duntley, S.Q., June 1971.

30. Naval Undersea Center Technical Publication 267, An Algorithm for Plotting Contours in Arbitrary Panar Regions, Smith, R.R., October 1971.

31. Jackson, J.D., Classical Electrodynamics, p. 643-647, Wiley, 1975.

32. Gagliardi, R.M. and Karp, S., Optical Communications, Wiley, 1976.

33. Yariv, A., Introduction to Optical Electronics, Holt, Rinehart and Winston, 1976.

34. Wylie, C.R., Advanced Engineering Math, p. 274, McGraw-Hill, 1966.

35. Deirmendjian, D., "Scattering and Polarization Properties of Water Clouds and Hazes in the Visible and Infrared," Applied Optics, V. 3, p. 187-196, February 1964.

36. Bucher, E.A. and Lerner, R.M., "Experiments on Light Pulse Communication and Propagation through Atmospheric Clouds," Applied Optics, V. 12, p. 2401-2414, October 1973.

37. Naval Electronics Laboratory Center Technical Note 3233, Experimental Test Plan to Investigate the Propagation of Blue/Green Radiation through Clouds, unpublished and tentative, October 1976.

38. Naval Electronics Laboratory Center Technical Report 1988, Extended Line-of-Sight Optical Communications Study, Mooridian, G.C., Adrian, N.J. and others, November 1976.

39. Naval Electronics Laboratory Center Technical Report 2022, Over-the-Horizon Optical Scatter Communications in the Marine Boundary Layer, Mooridian, G.C., Geller, M., and others, January 1977.

40. Moordaian, G.C. and others, "Blue/Green Pulse Propagation through Maritime Fogs," submitted for publication in Applied Optics.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria VA 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey CA 93940	2
3.	Department Chairman, Code 61 Department of Physics and Chemistry Naval Postgraduate School Monterey CA 93940	2
4.	Professor William M. Tolles, Code 61Tl Department of Physics and Chemistry Naval Postgraduate School Monterey CA 93940	10
5.	LT Dennis M. Junge 2705 N. Prospect Colorado Springs CO 80907	1
6.	Dr. Michael E. Neer Aeronautical Research Associates of Princeton, Inc. 50 Washington Road Princeton NJ 08540	1
7.	Professor A. Cooper, Code 61Cr Department of Physics and Chemistry Naval Postgraduate School Monterey CA 93940	1
8.	Dr. G.C. Mooradian, Code 8114 Naval Ocean Systems Center San Diego CA 92152	2
9.	Dr. E.A. Bucher Lincoln Laboratory Massachusetts Institute of Technology Lexington MS 02173	1
10.	LT Miles A. Millbach, USCG USCG Research and Development Center Avery Point Groton CT 06340	2